



ESCUELA SUPERIOR DE INGENIERÍA

INGENIERÍA TÉCNICA EN INFORMÁTICA DE SISTEMAS

Aplicación web para el análisis cuantitativo-cualitativo de actividad en wikis

Álvaro Almagro Doello

6 de octubre de 2015



ESCUELA SUPERIOR DE INGENIERÍA

INGENIERÍA TÉCNICA EN INFORMÁTICA DE SISTEMAS

Aplicación web para el análisis cuantitativo-cualitativo de actividad en wikis

- Departamento: Ingeniería Informática
- Directores: Antonio Balderas Alberico y Manuel Palomo Duarte
- Autor del proyecto: Álvaro Almagro Doello

Cádiz, 6 de octubre de 2015

Fdo: Álvaro Almagro Doello

Agradecimientos

Sin duda, el esfuerzo, las ganas, y el tiempo dedicados a este proyecto han sido cruciales para su llegada a buen puerto. Sin esas horas de máxima concentración, a veces robadas al sueño, esto no sería posible. Pero, incluso con todos esos factores, el entorno (y en concreto las personas) puede favorecer mucho a que un proyecto vaya como la seda, y mi entorno no podía ser más favorable. Quiero agradecer en primer lugar a Manuel Palomo, tutor del proyecto, que ha sabido llevarlo a la perfección, que no ha escatimado en ayudarme o en aportar comentarios constructivos y que ha tenido la paciencia necesaria para lidiar conmigo. Agradecer también a mi familia: a mis padres, Jose y Bea, que no han dejado de darme su apoyo en ningún momento; a mi hermana, Blanca, que siempre sabe como hacer que mis días de nubes sean algo más soleados; y a mi pareja, Sonia, que da sentido a todos mis logros. Y por último, quiero agradecer a mis compañeros de carrera, y en concreto a Jesús, sus aportes morales, intelectuales y de humor, que tanto han ayudado a mantener mi enfoque en el proyecto.

Índice general

Índice de figuras	III
Índice de cuadros	VII
I Prolegómeno	1
1. Introducción	3
1.1. Objetivo	3
1.2. Análisis de lo existente	3
1.3. Justificación	4
1.4. Estructura de la memoria	4
2. Fundamentos	5
2.1. Wikis	5
2.2. Evaluación cuantitativa. StatMediaWiki	5
2.3. Evaluación por pares. AssessMediaWiki	6
2.4. Uso de wikis en entornos educativos	7
2.5. Metodología de desarrollo	7
II Desarrollo	9
3. Análisis de requisitos	11
3.1. Objetivo principal	11
3.2. Entrevistas iniciales	11
3.3. Planificación temporal	12
3.4. Modelado	13
3.5. Especificación de requisitos	56
4. Diseño	59
4.1. Arquitectura del Sistema	59
4.2. Modelo de datos	59
4.3. Patrones de diseño (MVC)	63
4.4. Interfaz	64

5. Codificación	69
5.1. Entorno de programación	69
5.2. Lenguajes y herramientas	70
6. Pruebas	81
6.1. Pruebas unitarias en la API	81
6.2. Pruebas funcionales	82
 III Epílogo	 83
7. Manual de instalación	85
7.1. Pre-requisitos	85
7.2. Pasos de la instalación:	85
8. Manual de usuario	87
8.1. Sesión	87
8.2. Cambio de contraseña	87
8.3. Reportes	88
8.4. Configuración	90
9. Conclusiones	95
9.1. Conclusiones personales sobre el proyecto	95

Índice de figuras

2.1. Tabla de datos en StatMediaWiki	6
2.2. Gráfico de actividad por hora en StatMediaWiki	6
2.3. Definición de criterios en AssessMediaWiki	7
2.4. Resúmenes en AssessMediaWiki	8
3.1. Diagrama de clases	14
3.2. Caso de uso: Login de usuario	15
3.3. Caso de uso: Logout de usuario	16
3.4. Caso de uso: Cambiar contrasea	17
3.5. Caso de uso: Ver lista de reportes	17
3.6. Caso de uso: Añadir reporte	18
3.7. Caso de uso: Ver reporte general	19
3.8. Caso de uso: Ver reporte de alumno	20
3.9. Caso de uso: Ver reporte de página	21
3.10. Caso de uso: Ver reporte de categoría	22
3.11. Caso de uso: Ver reporte de grupo	23
3.12. Caso de uso: Umbrales de aviso	23
3.13. Caso de uso: Función de calidad	24
3.14. Caso de uso: Añadir wiki	25
3.15. Caso de uso: Borrar wiki	26
3.16. Caso de uso: Añadir grupo	27
3.17. Caso de uso: Asignar grupo	28
3.18. Caso de uso: Desasignar grupo	29
3.19. Caso de uso: Asignación automática de grupos	30
3.20. Caso de uso: Añadir fuente cualitativa	31
3.21. Caso de uso: Borrar fuente cualitativa	31
3.22. Caso de uso: Ver lista de usuarios	32
3.23. Caso de uso: Añadir usuario	33
3.24. Caso de uso: Borrar usuario	34
3.25. Caso de uso: Añadir función de calidad	34
3.26. Caso de uso: Añadir datos de contacto	35
3.27. Caso de uso: Ver información de contacto	36
3.28. Modelo de comportamiento: Login de usuario	37
3.29. Modelo de comportamiento: Logout de usuario	37
3.30. Modelo de comportamiento: Cambiar contrasea	38
3.31. Modelo de comportamiento: Ver listado de reportes	38
3.32. Modelo de comportamiento: Añadir reporte	39

3.33. Modelo de comportamiento: Ver reporte general	39
3.34. Modelo de comportamiento: Ver reporte de alumno	40
3.35. Modelo de comportamiento: Ver listado de página	40
3.36. Modelo de comportamiento: Ver reporte de categoría	41
3.37. Modelo de comportamiento: Ver reporte de grupo	41
3.38. Modelo de comportamiento: Umbrales de aviso	42
3.39. Modelo de comportamiento: Función de calidad	42
3.40. Modelo de comportamiento: Añadir wiki	43
3.41. Modelo de comportamiento: Borrar wiki	43
3.42. Modelo de comportamiento: Añadir grupo	44
3.43. Modelo de comportamiento: Asignar grupo	44
3.44. Modelo de comportamiento: Desasignar grupo	45
3.45. Modelo de comportamiento: Asignación automática de grupos	45
3.46. Modelo de comportamiento: Añadir fuente cualitativa	46
3.47. Modelo de comportamiento: Borrar fuente cualitativa	46
3.48. Modelo de comportamiento: Ver lista de usuarios	47
3.49. Modelo de comportamiento: Añadir usuario	47
3.50. Modelo de comportamiento: Borrar usuario	48
3.51. Modelo de comportamiento: Añadir función de calidad	48
3.52. Modelo de comportamiento: Añadir datos de contacto	49
3.53. Modelo de comportamiento: Ver información de contacto	49
4.1. Arquitectura del sistema	60
4.2. Modelo de datos	60
4.3. Modelo - Vista - Controlador	63
4.4. Mock-up de opciones	64
4.5. Mock-up de wikis	64
4.6. Mock-up de grupos	65
4.7. Mock-up de fuentes cualitativas	65
4.8. Mock-up de usuarios	66
4.9. Mock-up de reportes	66
4.10. Mock-up de popup de borrado	67
4.11. Mock-up de vista de reporte	67
4.12. Mock-up de contacto	68
4.13. Mock-up de menú de header	68
5.1. Entorno LAMP	69
5.2. Estructura de un repositorio Subversion	70
5.3. Conexiones entre bases de datos de CleverFigures	71
5.4. Estructura del servidor	73
5.5. Logotipo de JavaScript	73
5.6. Estructura base del cliente	74
5.7. Estructura del cliente con Flot	74
5.8. Estructura final del cliente	75
5.9. Estructura final de CleverFigures	76
5.10. Procesado secuencial	77
5.11. Procesado asíncrono	78
5.12. Estructura de la API de CleverFigures	79

8.1.	Pantalla de inicio de sesión	87
8.2.	Pantalla de cambio de contraseña	88
8.3.	Pantalla de listado de reportes	89
8.4.	Pantalla de creación de reportes	89
8.5.	Pantalla de visualización de reportes	90
8.6.	Pantalla de configuración	90
8.7.	Pantalla de configuración de wikis	91
8.8.	Pantalla de configuración de fuentes cualitativas	92
9.1.	Relación calidad - hora	96

Índice de cuadros

4.1. Tabla analysis (reportes)	61
4.2. Tabla color (fuentes cualitativas)	61
4.3. Tabla color_type (relación de fuentes cualitativas con tipos de fuentes cualitativas)	61
4.4. Tabla connection (credenciales de conexión para bases de datos de wikis y fuentes cualitativas)	61
4.5. Tabla groups (grupos de alumnos)	61
4.6. Tabla member (miembros de los grupos)	62
4.7. Tabla parameter (opciones de configuración de usuarios)	62
4.8. Tabla user (usuarios del sistema)	62
4.9. Tabla user_analysis (relaciones de usuarios con reportes)	62
4.10. Tabla user_color (relaciones de usuarios con fuentes cualitativas)	62
4.11. Tabla user_wiki (relaciones de usuarios con wikis)	62
4.12. Tabla wiki (wikis del sistema)	63

Parte I

Prolegómeno

Capítulo 1

Introducción

Hoy en día se hace casi imposible imaginar un sistema educativo sin ordenadores, Internet o la web. Para bien de muchos, y mal de algunos, la informatización ha supuesto grandes mejoras en las metodologías educativas, los medios de enseñanza y las formas de comunicación entre alumnos y docentes. Los continuos avances en el mundo de la informática y las telecomunicaciones hacen de la adaptación algo deseable y necesario. En este proyecto, queremos centrar nuestra atención en cómo la informática, y en concreto el uso de internet y la programación web, puede mejorar la forma en la que los docentes evalúan a sus alumnos. Hablaremos de cómo algunas metodologías, como la evaluación entre pares, han sido beneficiadas por la automatización derivada del desarrollo de aplicaciones web y expondremos los detalles de *CleverFigures*, una aplicación creada con el objetivo de mejorar notablemente los plazos de evaluación del profesorado en instituciones con grandes volúmenes de alumnos.

1.1. Objetivo

El objetivo principal de este proyecto es el desarrollo de una herramienta de análisis estadístico para entornos docentes, que combine tanto datos cuantitativos como cualitativos para dar una visión completa, fiable y fácilmente comprensible del trabajo realizado por los alumnos sobre la plataforma (en nuestro caso, el gestor de contenidos MediaWiki).

1.2. Análisis de lo existente

Previamente al desarrollo de este proyecto, existían varias herramientas que, de forma más o menos eficiente, conseguían resumir de forma gráfica el trabajo de los alumnos. Dos de ellas, en las cuales me he apoyado para desarrollar *CleverFigures*, fueron también desarrolladas bajo la tutela de la Universidad de Cádiz: *StatMediaWiki* y *AssessMediaWiki*. Estas herramientas eran, por entonces, unas de las más destacadas en lo que a análisis de sistemas MediaWiki con un enfoque docente se refiere. *StatMediaWiki* permite analizar y crear reportes numéricos objetivos de la evolución y el estado de la wiki, pero carece de información cualitativa o subjetiva que permita evaluar la calidad de dichos reportes. *AssessMediaWiki* es, por el contrario, puramente cualitativo. Permite que los alumnos puedan evaluarse entre ellos, y que el personal docente pueda validar dichas evaluaciones mediante hetero y auto-evaluaciones. Una vez poblado el sistema con estas evaluaciones, es fácil saber cómo de bien lo ha hecho un alumno. Por contra, tenemos una carencia: 'cuánto' ha hecho. La práctica hasta ahora consistía en combinar, de forma manual, los datos provenientes de estas dos herramientas para obtener un informe completo.

1.3. Justificación

Al analizar las herramientas previamente existentes y el uso que se les daba, resultaba obvio que era posible automatizar el proceso final de combinación de datos provenientes de ambas fuentes para obtener directamente un informe completo. La síntesis manual de dichos datos era una tarea imprescindible para obtener informes fiables, pero a la vez, que consumía esfuerzos y tiempo, recursos que como veremos más adelante en esta memoria, era posible ahorrar.

1.4. Estructura de la memoria

En las próximas páginas de esta parte de la memoria, encontraremos el capítulo de Fundamentos, en el cual se dará una explicación detallada de las bases del proyecto. En la siguiente parte, Desarrollo, podremos leer sobre las distintas fases del desarrollo del proyecto, como son el análisis de requisitos, el diseño, la codificación y las pruebas del software resultante. En la última parte de la memoria, Epílogo, se incluyen los manuales del programa y las conclusiones del proyecto, así como las referencias, la bibliografía y la información de licencia.

Capítulo 2

Fundamentos

2.1. Wikis

CleverFigures basa su funcionamiento en el modelo de datos de las wikis, en nuestro caso, gestionadas por el CMS MediaWiki[9], uno de los más usados y complejos. Una wiki es, según Wikipedia (que en sí es un buen ejemplo de wiki), 'el nombre que recibe un sitio web cuyas páginas pueden ser editadas directamente desde el navegador, donde los usuarios crean, modifican o eliminan contenidos que, generalmente, comparten'. CleverFigures usa las wikis como fuente de datos numéricos para la elaboración de sus informes mediante una API propia, que también pone a disposición de los desarrolladores. Nuestra intención, al usar CleverFigures sobre una wiki, es la de mostrar de forma comprensible y resumida los datos que, en nuestro caso, los alumnos han aportado a la misma. Estos datos serán combinados con información cualitativa para elaborar informes enriquecidos de forma totalmente automática. De esta forma, pretendemos facilitar el trabajo de los docentes a la hora de obtener evaluaciones del alumnado [16, 17].

2.2. Evaluación cuantitativa. StatMediaWiki

El trabajo de los alumnos sobre las wikis genera gran cantidad de datos con gran valor a la hora de evaluar. Parámetros como la cantidad de texto aportada, la actividad diaria, el cumplimiento de los plazos, etc pueden resultar claves a la hora de evaluar a un alumno si nos basamos en su trabajo en la wiki. Las medias, modas y demás indicadores estadísticos son en muchos casos la base objetiva de las evaluaciones. En asignaturas con grandes volúmenes de alumnos estos datos resultan, en la mayoría de los casos, difíciles de manejar, sintetizar y clasificar. Las wikis realizan un gran papel al poner a disposición, tanto de alumnos como docentes, un medio en el que centralizar todos estos datos y clasificarlos de forma lógica y jerárquica. Pero incluso con una wiki, analizar todos estos datos y dar una visión clara de su significado no es fácil. Existen herramientas para sintetizar todo el trabajo de forma gráfica, y disponer de un resumen visual de los datos numéricos que en ella se alojan. StatMediaWiki [21] nació con dicho objetivo. StatMediaWiki es una herramienta desarrollada por la Universidad de Cádiz y predecesora de CleverFigures, que nos permite obtener informes numéricos y objetivos del trabajo realizado en la wiki. Estos informes son generados en páginas HTML y pueden ser consultados por alumno, página o categoría de la wiki.

#	User	Edits	%	Edits in articles	%	Bytes	%	Bytes in articles	%	Uploads	%
1	Anon1	220	14.3%	203	16.1%	41442	9.6%	40424	10.3%	3	7.7%
2	Anon2	177	11.5%	136	10.8%	32488	7.5%	22498	5.8%	4	10.3%
3	Anon3	109	7.1%	80	6.3%	92102	21.3%	88195	22.6%	2	5.1%
4	Anon4	106	6.9%	93	7.4%	24640	5.7%	20820	5.3%	1	2.6%
5	Anon5	102	6.6%	79	6.3%	20704	4.8%	20373	5.2%	10	25.6%
6	Anon6	100	6.5%	92	7.3%	37305	8.6%	36424	9.3%	2	5.1%
7	Anon7	83	5.4%	68	5.4%	18712	4.3%	17344	4.4%	6	15.4%
8	Anon8	76	4.9%	71	5.6%	17212	4.0%	16956	4.3%	2	5.1%

Figura 2.1: Tabla de datos en StatMediaWiki

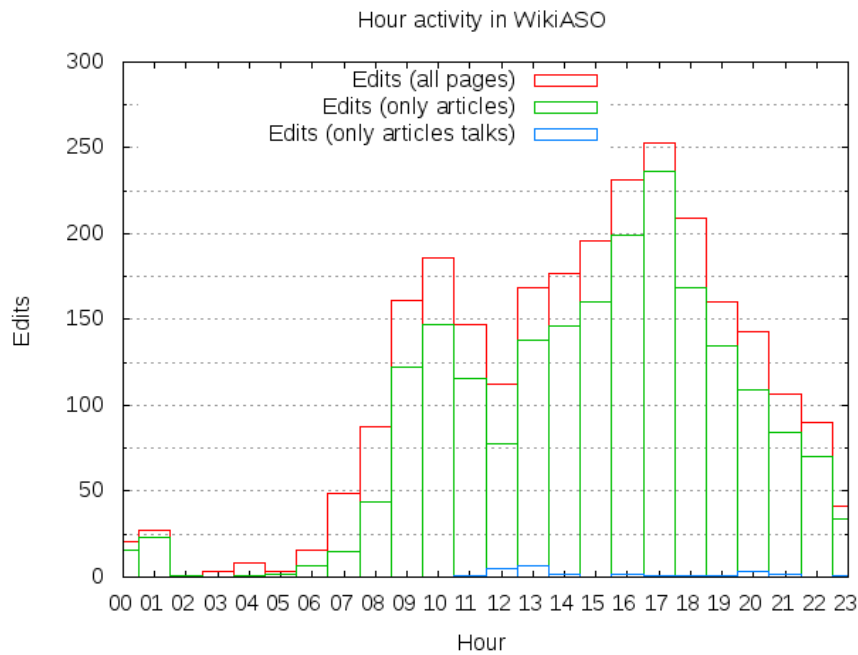


Figura 2.2: Gráfico de actividad por hora en StatMediaWiki

2.3. Evaluación por pares. AssessMediaWiki

Las wikis son un gran almacén de datos y una indudable fuente de conocimiento, pero esto solo es cierto si las aportaciones que se han hecho a la misma son filtradas y evaluadas de forma exhaustiva por el resto de la comunidad [18]. En el ámbito de la enseñanza, esto se consigue hacer de manera escalable a la vez que formativa, mediante la evaluación por pares. La evaluación por pares permite a los alumnos comportarse como una comunidad y llevar a cabo esta tarea de evaluación que antes mencionábamos. Cada alumno evaluará el trabajo de uno o varios compañeros, y dará su opinión sobre dicha evaluación. Esto, con un número alto de alumnos, medirá la calidad del trabajo que finalmente es almacenado en la wiki, y el grado de adquisición de competencias del alumnado. Con este objetivo en mente, nació AssessMediaWiki [20]. AssessMediaWiki es una aplicación web de código abierto que, al conectarse a una instalación MediaWiki, proporciona procedimientos de autoevaluación, hetero evaluación y evaluación entre

iguales, a la vez que mantiene información sobre esas evaluaciones. Los supervisores pueden obtener informes que ayudan en la evaluación de los estudiantes. AssessMediaWiki es la principal fuente de datos cualitativos de CleverFigures, lo que permite añadir a los informes de evaluación una síntesis de la calidad de los trabajos registrados por los alumnos.

Revision	Grade	Description
<input type="checkbox"/> Justificación Se justifica la necesidad de la migración	0	
<input type="checkbox"/> Sala anterior Diseño de la sala de servidores (antes de migrar)	0	
<input type="checkbox"/> Sala después Diseño de la sala de servidores (después de migrar)	0	
<input type="checkbox"/> Servidor físico Hardware y software del servidor(es) físico	0	
<input type="checkbox"/> Servidor virtual Hardware y software del servidor(es) virtual	0	
<input type="checkbox"/> Red Conexiones básica de red	0	

Figura 2.3: Definición de criterios en AssessMediaWiki


2.4. Uso de wikis en entornos educativos

Muchas empresas y universidades han optado por el uso de las wikis como modo de divulgación y distribución de información y conocimiento [19]. Las ventajas que ofrecen son numerosas, entre las cuales destacan su estructura flexible, que facilita la edición colaborativa, la transparencia y registro de todo lo sucedido en el entorno, y su organización por categorías y páginas temáticas, que hace que encontrar lo que se busca resulte trivial. En la universidad, en concreto, las wikis otorgan un manera inmejorable de favorecer la colaboración entre alumnos y de almacenar y hacer evolucionar positivamente el conocimiento que estos van adquiriendo a lo largo de su paso por la misma [22]. El resultado del buen uso de una wiki será una base de conocimiento centralizado, disponible para todos y mejorable por todos.[1, 2]

2.5. Metodología de desarrollo

Como metodología de desarrollo seguiremos el modelo en cascada. Este modelo consiste en dividir el desarrollo del proyecto en distintas fases fundamentales. En nuestro caso, como podemos ver en el diagrama de Gantt adjunto en la sección de planificación, las fases son: entrevistas iniciales, análisis, diseño, codificación, calidad y documentación.

AssessMediaWiki
AssessMediaWiki - Google Chrome
localhost/~jose/assessmediawiki/assessmediawiki/index.php/evaluar/mostrar_evalua


AssessMediaWiki

Assess
My assessments
Logout

Summary of assessment

- Author: Alguar
- Revision ID: 2013
- Revision link: [url](#)

Revision	Grade	Description
Formación	7	Añade introducción a los cursos de formación
Escritura	9	Hace uso de formato wiki más adecuado

If you don't agree with your evaluation, you may reply [clicking here](#).

Summary of reply 1

- Author: Alguar
- Revisor: Alguar
- Revision ID: 2013
- Revision link: [url](#)

Revision	Grade	Description
Justificación	3	
Sala después	9	

Summary of reply 2

- Author: Alguar
- Revisor: Alguar
- Revision ID: 2013

Revision	Grade	Description
Servidor físico	9	

Figura 2.4: Resúmenes en AssessMediaWiki

Parte II

Desarrollo

Capítulo 3

Análisis de requisitos

3.1. Objetivo principal

El problema a resolver era claro: disponíamos de las herramientas para recopilar de forma organizada el trabajo de los alumnos (wikis con datos numéricos) y teníamos una herramienta que nos permitía aplicar la evaluación por pares a dicho trabajo (AssessMediaWiki con datos cualitativos), pero no existía ninguna herramienta que analizara y sintetizara tanto el trabajo como las evaluaciones y pusiera a nuestra disposición un resumen claro que combinase datos numéricos y cualitativos que nos permitiera evaluar fácilmente al alumnado. Hasta ahora, el personal docente debía revisar el trabajo y, manualmente, combinarlo con las evaluaciones y comentarios almacenados en AssessMediaWiki para poder generar una evaluación definitiva del alumno.

3.2. Entrevistas iniciales

Para acordar de qué forma se abordaría el problema, y cómo se implementaría su solución, se realizaron varias entrevistas con el personal docente, y más concretamente, con los profesores involucrados en asignaturas que usaban MediaWiki como base para el desarrollo del trabajo de sus alumnos. Estas entrevistas se sucedieron en las dos primeras semanas después del comienzo del proyecto, y culminaron con una propuesta que se explica en el siguiente apartado.

3.2.1. Propuesta detallada

Una vez finalizadas las entrevistas iniciales, se llegó a un acuerdo sobre las tecnologías, procedimientos y herramientas a utilizar durante el proyecto, y por supuesto, qué forma debía tener el producto resultante. El sistema debía ofrecer una interfaz web sobre la cual el personal docente pudiera trabajar. Esto restringe el paradigma a la programación web. La razón de esto es facilitar al máximo el acceso al sistema, desde cualquier lugar y sin restricciones impuestas por el dispositivo o sistema operativo desde el que se trabaja, más allá de los límites acordados. Nuestra tarea era, entonces, desarrollar una aplicación web que permitiera analizar el trabajo de los alumnos de forma tanto cuantitativa como cualitativa, que mostrara esos análisis de forma gráfica y dinámica, y que mejorara el rendimiento de las herramientas existentes, que, en algunos casos, podían tardar horas en ofrecer un resultado. También era necesario poder guardar los reportes y facilitar su administración y exportación a otras herramientas. Sobre los lenguajes de programación y metodologías a utilizar, se dejaron a elección del desarrollador, con dos restricciones: el uso de PHP y CodeIgniter (sobre los que se había desarrollado también AssessMediaWiki) para el back-end

de la aplicación y la obligación de generar los gráficos de forma dinámica, y por consiguiente, de añadir código en el lado del cliente. De todo esto hablaremos más detalladamente en el apartado de codificación.

3.3. Planificación temporal

Se realizó una planificación inicial para el proyecto. El desarrollo se demoró por diversos motivos de índole personal (cambio de residencia del alumno, así como de su situación laboral), por lo que no fue posible cumplir con la planificación inicial. Ambas situaciones han sido reflejadas en los correspondientes diagramas de Gantt que podemos ver en el apéndice de la página 146.

3.3.1. Costes

El proyecto fue realizado por el alumno en un programa de prácticas de empresa preestablecido, por lo que el presupuesto ya estaba cerrado. Aunque el presupuesto final no fuera flexible, sí que podemos calcular el sobre coste por el retraso en una simulación.

Coste estimado - condiciones del programa de prácticas

- Número de personas trabajando en el proyecto: 1.
- Salario mensual bruto: 350€.
- Duración de las prácticas: 7 meses.

Como podemos ver, el coste total del proyecto asciende a

$$1350€ \times 7 = 2450€ \quad (3.1)$$

que fue la cantidad finalmente retribuida.

Coste real - coste real simulado dado el retraso sufrido

- Número de personas trabajando en el proyecto: 1.
- Salario mensual bruto: 350€.
- Duración de las prácticas: 6 meses.
- Retraso en el desarrollo fuera del programa de prácticas (sin contar los meses de inactividad): 9 meses.

Si calculamos el coste en este caso obtenemos

$$1350€ \times 15 = 5250€ \quad (3.2)$$

Podemos ver que el proyecto tuvo un sobre coste (simulado) de

$$5250€ - 2450€ = 2800€ \quad (3.3)$$

3.4. Modelado

3.4.1. Diagrama de clases

A continuación se describen las entidades usadas para modelar la aplicación.

Usuario (user)

Es la entidad que representa a los usuarios del sistema. En nuestro caso, podrán ser administradores o profesores.

Wiki (wiki)

Representa a las wikis que se conectarán con CleverFigures y que servirán de fuente de datos principal para los reportes.

Fuente cualitativa (color)

Representa a las fuentes cualitativas que los usuarios conectarán al sistema para enriquecer los reportes.

Reporte (analysis)

Representa a los reportes que generarán los usuarios en la aplicación.

Conexión (connection)

Representa las conexiones de bases de datos a bajo nivel que las wikis y las fuentes cualitativas necesitan.

Parámetro (parameter)

Representa las opciones de personalización que el usuario puede configurar.

Grupo (group)

Representa los grupos de alumnos que se pueden formar a partir de los usuarios de una wiki.

Miembro (member)

Representa a los alumnos pertenecientes a los grupos de la entidad anterior.

3.4.2. Modelo de casos de uso

En este apartado describiremos todos los casos de uso que el sistema contempla. Cada caso de uso va acompañado por su esquema UML correspondiente.

Actores disponibles

Administrador: Representa a un usuario con permisos totales de configuración del sistema.

Profesor: Representa a un profesor registrado en el sistema.

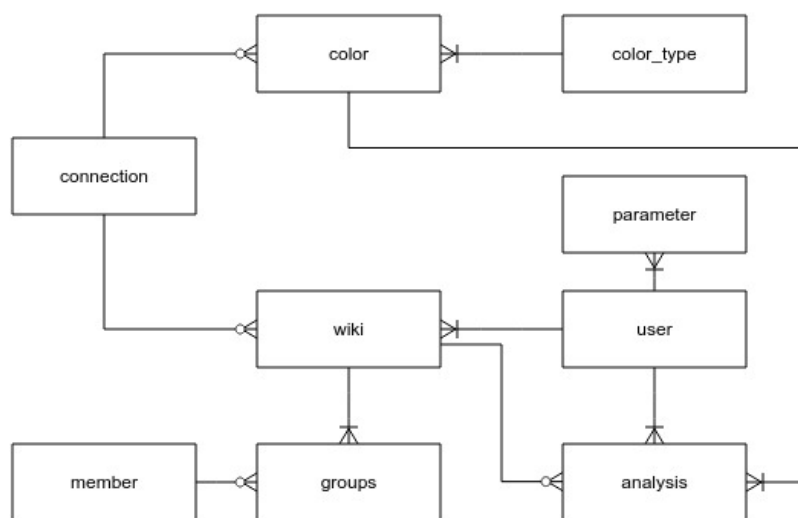


Figura 3.1: Diagrama de clases

Sesión - Login

Descripción: Login de un usuario.

Actores: Administrador, Profesor.

Resumen: Cuando el usuario abre CleverFigures, es necesario que introduzca sus credenciales de acceso (usuario y contraseña) para poder acceder al sistema.

Escenario principal:

1. El usuario desea acceder a CleverFigures y abre la aplicación en su navegador.
2. El usuario introduce su usuario y su contraseña y pulsa Submit.
3. El sistema valida sus credenciales y le permite acceder.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 3a. El usuario no existe y el sistema informa al usuario de que sus credenciales de acceso no son correctas.
- 3b. La contraseña no es correcta y el sistema informa al usuario de que sus credenciales de acceso no son correctas.
- 3c. Alguno de los campos está vacío y el sistema informa al usuario de que sus credenciales de acceso no son correctas.

Sesión - Salir

Descripción: Logout de un usuario.

Actores: Administrador, Profesor.

Resumen: Cuando el usuario quiere salir de CleverFigures, pulsa el botón de salir y abandona el

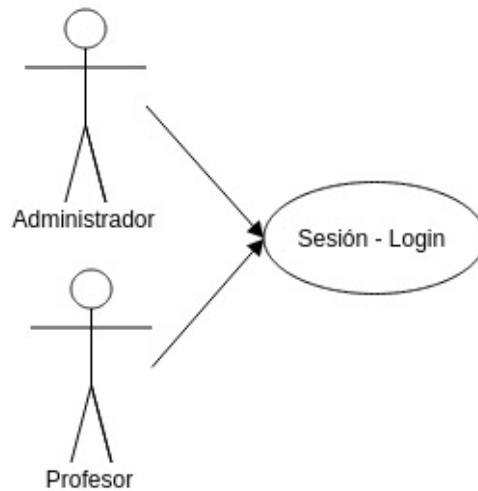


Figura 3.2: Caso de uso: Login de usuario

sistema.

Escenario principal:

1. El usuario desea salir de CleverFigures.
2. El usuario pulsa el botón Salir.
3. El sistema elimina su sesión y lleva al usuario a la página de Login.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

Cuenta - Cambiar contraseña

Descripción: Cambio de contraseña de acceso.

Actores: Administrador, Profesor.

Resumen: Un usuario realiza un cambio de contraseña para las próximas sesiones.

Escenario principal:

1. El usuario desea cambiar su contraseña y pulsa el botón “Cambiar contraseña”.
2. El usuario introduce su contraseña antigua, su nueva contraseña y la confirmación de la misma. Pulsa “Cambiar contraseña”.
3. El sistema valida que su antigua contraseña es correcta y, si la nueva contraseña coincide con la confirmación, realiza el cambio en la base de datos.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier

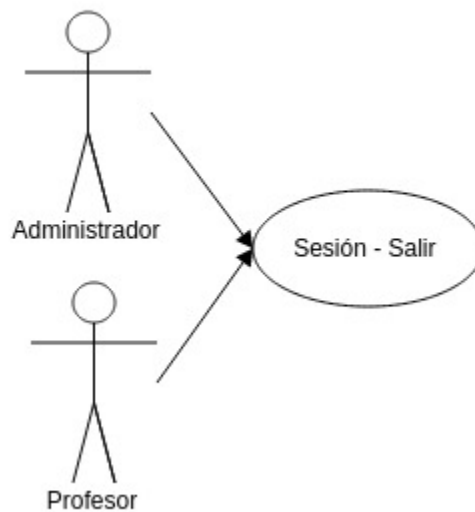


Figura 3.3: Caso de uso: Logout de usuario

momento.

3a. La contraseña antigua no es correcta. El sistema informa al usuario del problema con un mensaje de error y no se realizan cambios.

3b. La nueva contraseña no coincide con la confirmación. El sistema informa al usuario del problema y no se realizan cambios.

3c. Alguno de los campos está vacío y el sistema informa al usuario de que todos los campos son obligatorios. No se realizan cambios.

Reportes - Ver lista de reportes

Descripción: El usuario visualiza una lista de todos los reportes creados hasta el momento.

Actores: Administrador, Profesor.

Resumen: Un usuario desea ver una lista de todos los reportes que ha generado hasta el momento.

Escenario principal:

1. El usuario desea ver todos los reportes que ha creado hasta ahora y pulsa "Reportes".
2. El sistema muestra una lista de todos los reportes que el usuario ha registrado hasta el momento.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

Reportes - Añadir reporte

Descripción: Creación de un nuevo reporte.

Actores: Administrador, Profesor.

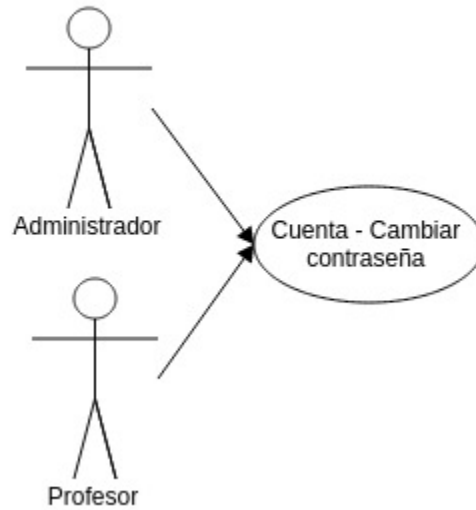


Figura 3.4: Caso de uso: Cambiar contraseña

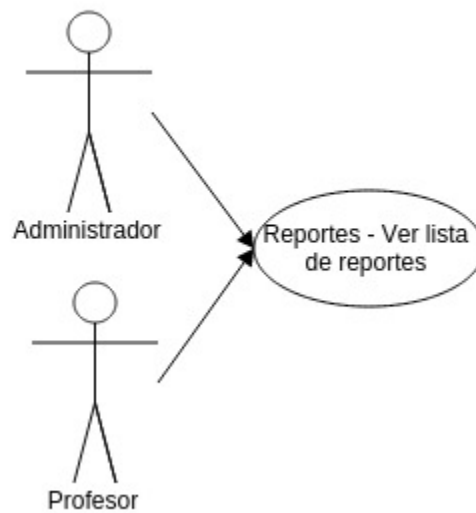


Figura 3.5: Caso de uso: Ver lista de reportes

Resumen: Un usuario genera un nuevo reporte en el sistema.

Escenario principal:

1. El usuario desea crear un nuevo reporte, pulsa el botón “Reportes” y después “Nuevo reporte”.

2. El usuario especifica qué wiki y fuente cualitativa de las disponibles quiere usar para generar el nuevo reporte.
3. El sistema genera el nuevo reporte para la fecha actual. El nuevo reporte aparece listado en el listado de reportes.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 2a. El usuario no desea usar una fuente cualitativa y deja dicho campo vacío. En este caso el reporte se generará sin datos cualitativos.
- 3a. El campo wiki está vacío y el sistema no puede generar el reporte. Se informa al usuario del problema.

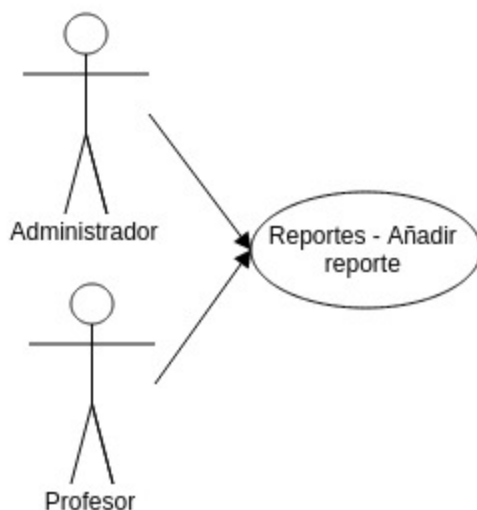


Figura 3.6: Caso de uso: Añadir reporte

Reportes - Ver reporte - General

Descripción: Visualización de reporte general de una wiki.

Actores: Administrador, Profesor.

Resumen: Un usuario abre un reporte en el listado de reportes.

Escenario principal:

1. El usuario desea ver un reporte general de la wiki y abre el reporte en el listado de reportes.
2. Se muestra el reporte con información general de toda la wiki, así como la información cualitativa que se haya generado para la misma.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier

momento.

2a. La wiki no dispone de información cualitativa. En este caso, los gráficos correspondientes a dicha información no aparecen en el reporte.

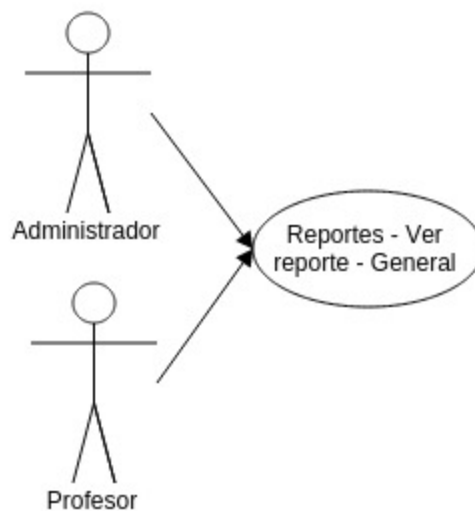


Figura 3.7: Caso de uso: Ver reporte general

Reportes - Ver reporte - Reporte de alumno

Descripción: Visualización de reporte de un alumno de una wiki.

Actores: Administrador, Profesor.

Resumen: Un usuario abre un reporte en el listado de reportes y accede a la información específica de un alumno.

Escenario principal:

1. El usuario desea ver un reporte de un alumno y abre el reporte de la wiki del alumno en cuestión.
2. Se muestra el reporte con información general de toda la wiki, el usuario navega hasta la tabla de usuarios y hace click en un alumno.
3. Se muestra el reporte correspondiente a la información cuantitativa y cualitativa relacionada con dicho alumno.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

3a. El reporte no se generó con información cualitativa. En este caso, los gráficos cualitativos no estarán disponibles en el reporte.

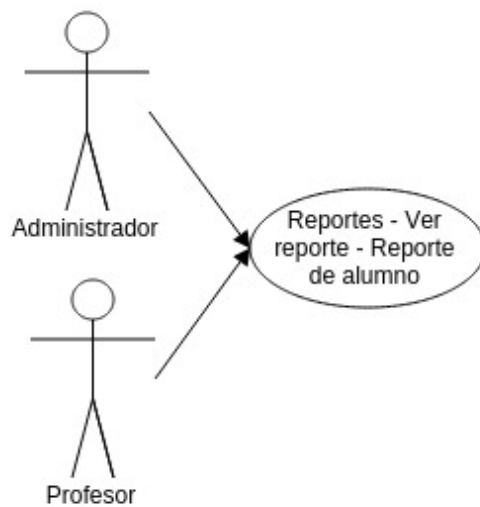


Figura 3.8: Caso de uso: Ver reporte de alumno

Reportes - Ver reporte - Reporte de página

Descripción: Visualización de reporte de una página de una wiki.

Actores: Administrador, Profesor.

Resumen: Un usuario abre un reporte en el listado de reportes y accede a la información específica de una página.

Escenario principal:

1. El usuario desea ver un reporte de página y abre el reporte de la wiki de la página en cuestión.
2. Se muestra el reporte con información general de toda la wiki, el usuario navega hasta la tabla de páginas y hace click en una página.
3. Se muestra el reporte correspondiente a la información cuantitativa y cualitativa relacionada con dicha página de la wiki.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

3a. El reporte no se generó con información cualitativa. En este caso, los gráficos cualitativos no estarán disponibles en el reporte.

Reportes - Ver reporte - Reporte de categoría

Descripción: Visualización de reporte de una categoría de una wiki.

Actores: Administrador, Profesor.

Resumen: Un usuario abre un reporte en el listado de reportes y accede a la información específica

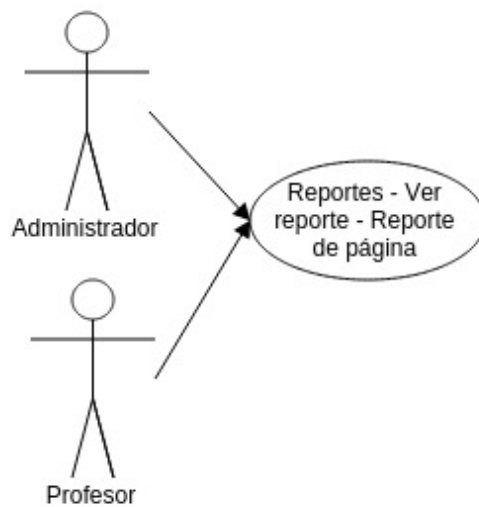


Figura 3.9: Caso de uso: Ver reporte de página

de una categoría.

Escenario principal:

1. El usuario desea ver un reporte de categoría y abre el reporte de la wiki de la categoría en cuestión.
2. Se muestra el reporte con información general de toda la wiki, el usuario navega hasta la tabla de categorías y hace click en una categoría.
3. Se muestra el reporte correspondiente a la información cuantitativa y cualitativa relacionada con dicha categoría.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 3a. El reporte no se generó con información cualitativa. En este caso, los gráficos cualitativos no estarán disponibles en el reporte.

Reportes - Ver reporte - Reporte de grupo

Descripción: Visualización de reporte de un grupo de una wiki.

Actores: Administrador, Profesor.

Resumen: Un usuario abre un reporte en el listado de reportes y accede a la información específica de un grupo de alumnos.

Escenario principal:

1. El usuario desea ver un reporte de grupo y abre el reporte de la wiki del grupo en cuestión.

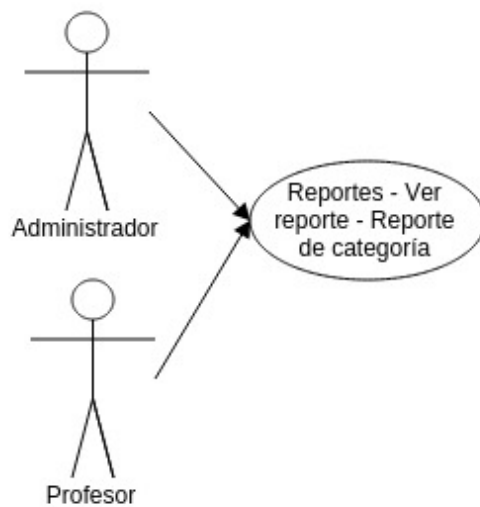


Figura 3.10: Caso de uso: Ver reporte de categoría

2. Se muestra el reporte con información general de toda la wiki, el usuario navega hasta la tabla de grupos y hace click en un grupo.
3. Se muestra el reporte correspondiente a la información cuantitativa y cualitativa relacionada con dicho grupo.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 3a. El reporte no se generó con información cualitativa. En este caso, los gráficos cualitativos no estarán disponibles en el reporte.

Configurar - Opciones - Umbrales de aviso

Descripción: Configuración de umbrales de aviso para los reportes.

Actores: Administrador, Profesor.

Resumen: Un usuario desea personalizar sus umbrales de aviso y accede a la configuración de los mismos.

Escenario principal:

1. El usuario desea personalizar sus umbrales de aviso y accede a la configuración de los mismos pulsando el botón “Configuración”.
2. En la sección de “Umbrales de aviso” el usuario establece los mínimos y máximos que desea tener en cuenta en sus reportes. Después pulsa el botón “Guardar”
3. El sistema registra estos nuevos umbrales y los tiene en cuenta a la hora de mostrar los reportes.

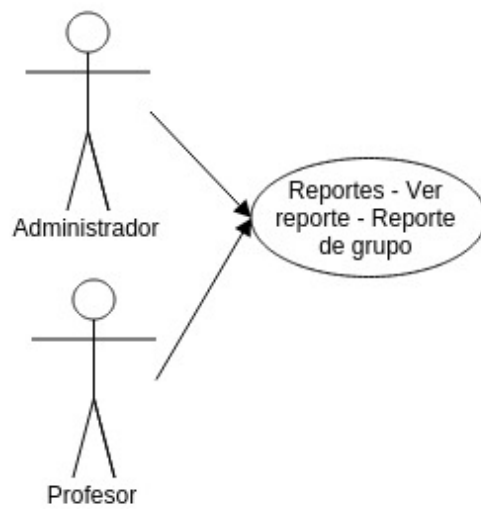


Figura 3.11: Caso de uso: Ver reporte de grupo

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

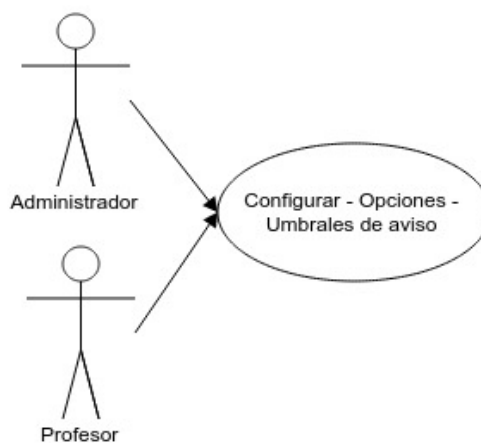


Figura 3.12: Caso de uso: Umbrals de aviso

Configurar - Opciones - Función de calidad

Descripción: Configuración de función de calidad para los reportes.

Actores: Administrador, Profesor.

Resumen: Un usuario desea personalizar su función de calidad y accede a la configuración de la misma.

Escenario principal:

1. El usuario desea personalizar su función de calidad y accede a la configuración de la misma pulsando el botón “Configuración”.
2. En la sección de “Opciones de fuentes cualitativas” el usuario establece la función de calidad que desea tener en cuenta en sus reportes. Después pulsa el botón “Guardar”
3. El sistema registra esta nueva función y la tiene en cuenta a la hora de mostrar los reportes.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

3a. La función de calidad no ha sido registrada previamente por el administrador en el sistema, por lo que su uso no está permitido. Se informa al usuario mediante un mensaje de error.

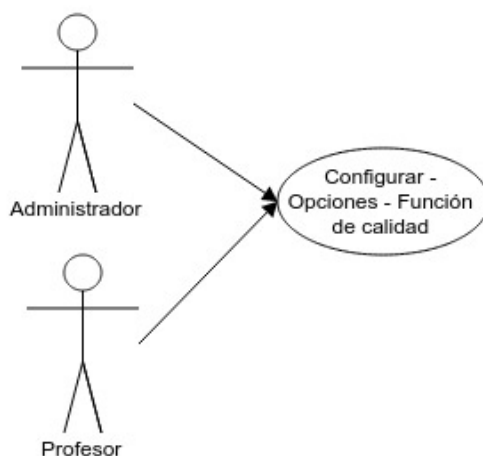


Figura 3.13: Caso de uso: Función de calidad

Configurar - Wikis - Añadir wiki

Descripción: Adición de una nueva wiki.

Actores: Administrador, Profesor.

Resumen: Un usuario desea añadir una nueva wiki para posibilitar la generación de informes para la misma.

Escenario principal:

1. El usuario desea añadir una nueva wiki. Accede al apartado de configuración mediante el botón “Configuración”.
2. El usuario hace click en la pestaña “Wikis” y después en “Añadir wiki”
3. El usuario introduce los datos de conexión con la base de datos de la wiki y pulsa “Guardar”
4. El sistema registra la nueva wiki. Estará disponible para generar nuevos reportes.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 4a. Alguno de los campos está vacío. El sistema avisa al usuario mediante un mensaje de error.
- 4b. Alguno de los campos no es correcto y el sistema no puede conectar con la base de datos de la wiki. El sistema informa al usuario mediante un mensaje de error.

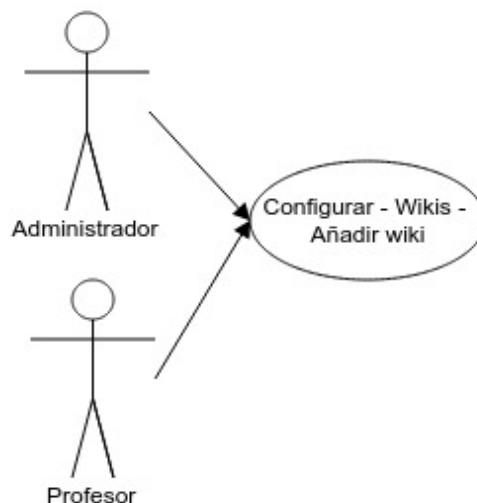


Figura 3.14: Caso de uso: Añadir wiki

Configurar - Wikis - Borrar wiki

Descripción: Borrado de una wiki.

Actores: Administrador, Profesor.

Resumen: Un usuario desea borrar una wiki del sistema.

Escenario principal:

1. El usuario desea borrar una wiki. Accede al apartado de configuración mediante el botón “Configuración”.
2. El usuario hace click en la pestaña “Wikis” y después en el icono de borrar en la wiki en cuestión.

3. El sistema muestra un pop-up de confirmación.
4. El usuario confirma el borrado. La wiki es borrada del sistema.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 4a. El usuario decide cancelar el borrado mediante el botón “Cancelar”. En este caso el sistema no realiza ningún cambio en la base de datos y la wiki sigue estando disponible.

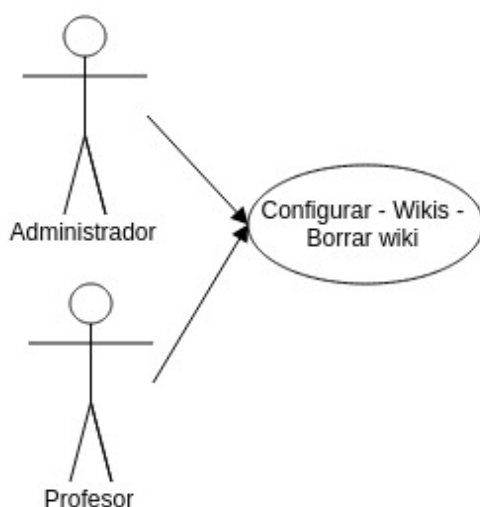


Figura 3.15: Caso de uso: Borrar wiki

Configurar - Wikis - Gestionar grupos - Añadir grupo

Descripción: Adición de un grupo.

Actores: Administrador, Profesor.

Resumen: Un usuario desea añadir un grupo a una wiki del sistema.

Escenario principal:

1. El usuario desea añadir un grupo a una wiki. Accede al apartado de configuración mediante el botón “Configuración”.
2. El usuario hace click en la pestaña “Wikis” y después en el icono de gestionar grupos en la wiki en cuestión.
3. El sistema muestra la pantalla de gestión de grupos.
4. El usuario pulsa “Añadir grupo”.
5. El sistema muestra un pop-up en el que el usuario puede introducir el nombre del grupo.

6. El usuario introduce el nombre del grupo y pulsa “Guardar”.
7. El nuevo grupo queda registrado en el sistema.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 6a. El usuario pulsa “Cancelar”. No se registra ningún grupo.
- 6b. El usuario pulsa “Guardar” sin escribir un nombre y el sistema informa del error mediante un mensaje. No se registra ningún grupo.

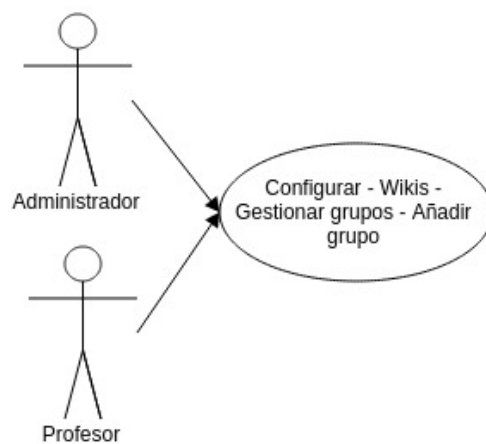


Figura 3.16: Caso de uso: Añadir grupo

Configurar - Wikis - Gestionar grupos - Asignar grupo

Descripción: Asignación de un grupo.

Actores: Administrador, Profesor.

Resumen: Un usuario desea asignar un grupo a un alumno de una wiki del sistema.

Escenario principal:

1. El usuario desea asignar un grupo a un alumno de una wiki. Accede al apartado de configuración mediante el botón “Configuración”.
2. El usuario hace click en la pestaña “Wikis” y después en el icono de gestionar grupos en la wiki en cuestión.
3. El sistema muestra la pantalla de gestión de grupos.
4. El usuario navega hasta el alumno en cuestión y pulsa “Añadir grupo”.
5. El sistema muestra un pop-up en el que el usuario elige el grupo a asignar y pulsa “Asignar”.
6. El sistema asigna el grupo del alumno de la wiki.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

5b. El usuario pulsa “Cancelar” y la asignación no se produce.

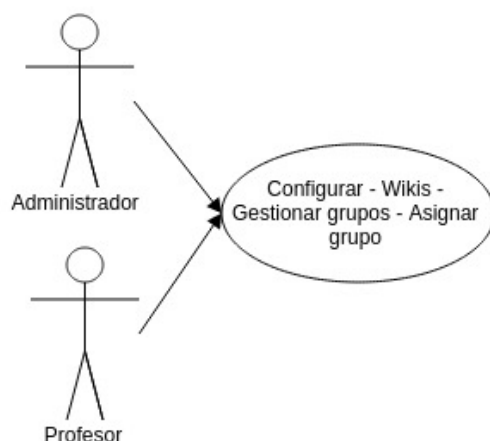


Figura 3.17: Caso de uso: Asignar grupo

Configurar - Wikis - Gestionar grupos - Desasignar grupo

Descripción: Desasignación de un grupo.

Actores: Administrador, Profesor.

Resumen: Un usuario desea desasignar un grupo de un alumno de una wiki del sistema.

Escenario principal:

1. El usuario desea desasignar un grupo de un alumno de una wiki. Accede al apartado de configuración mediante el botón “Configuración”.
2. El usuario hace click en la pestaña “Wikis” y después en el icono de gestionar grupos en la wiki en cuestión.
3. El sistema muestra la pantalla de gestión de grupos.
4. El usuario navega hasta el alumno en cuestión y pulsa el aspa roja junto al grupo.
5. El sistema desasigna el grupo del alumno de la wiki.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

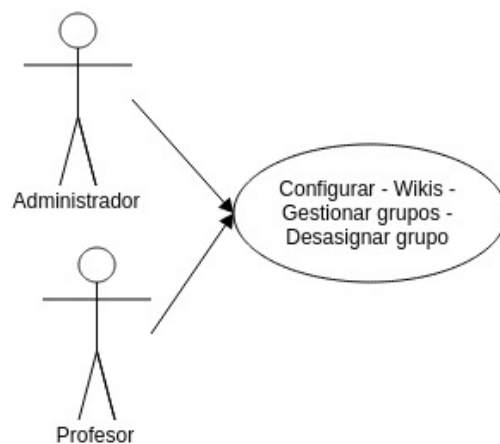


Figura 3.18: Caso de uso: Desasignar grupo

Configurar - Wikis - Gestionar grupos - Asignación automática de grupos

Descripción: Asignación automática de grupos.

Actores: Administrador, Profesor.

Resumen: Un usuario desea asignar grupos de forma automática a los alumnos basándose en las páginas en las que estos han trabajado.

Escenario principal:

1. El usuario desea asignar grupos a los alumnos de una wiki de forma automática. Accede al apartado de configuración mediante el botón “Configuración”.
2. El usuario hace click en la pestaña “Wikis” y después en el icono de gestionar grupos en la wiki en cuestión.
3. El sistema muestra la pantalla de gestión de grupos.
4. El usuario pulsa el botón “Agrupar por página”.
5. El sistema asigna grupos a los alumnos de forma automática, basándose en las páginas en las que estos han trabajado.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

Configurar - Fuentes cualitativas - Añadir fuente cualitativa

Descripción: Adición de una nueva fuente cualitativa.

Actores: Administrador, Profesor.

Resumen: Un usuario desea añadir una nueva fuente cualitativa para posibilitar la generación de informes usando la misma.

Escenario principal:

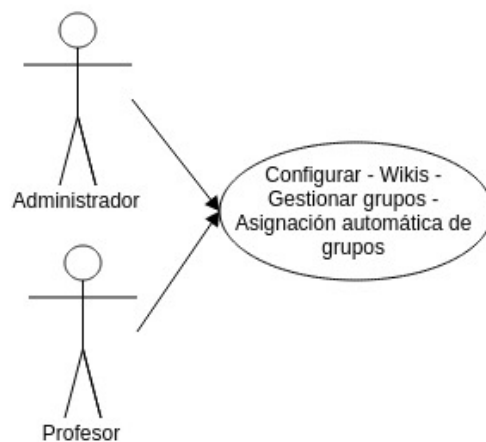


Figura 3.19: Caso de uso: Asignación automática de grupos

1. El usuario desea añadir una nueva fuente cualitativa. Accede al apartado de configuración mediante el botón “Configuración”.
2. El usuario hace click en la pestaña “Fuentes cualitativas” y después en “Añadir fuente cualitativa”
3. El usuario introduce los datos de conexión con la base de datos de la fuente cualitativa y pulsa “Guardar”
4. El sistema registra la nueva fuente cualitativa. Estará disponible para generar nuevos reportes.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 4a. Alguno de los campos está vacío. El sistema avisa al usuario mediante un mensaje de error.
- 4b. Alguno de los campos no es correcto y el sistema no puede conectar con la base de datos de la fuente cualitativa. El sistema informa al usuario mediante un mensaje de error.

Configurar - Fuentes cualitativas - Borrar fuente cualitativa

Descripción: Borrado de una fuente cualitativa.

Actores: Administrador, Profesor.

Resumen: Un usuario desea borrar una fuente cualitativa del sistema.

Escenario principal:

1. El usuario desea borrar una fuente cualitativa. Accede al apartado de configuración mediante el botón “Configuración”.
2. El usuario hace click en la pestaña “Fuentes cualitativas” y después en el icono de borrar en la fuente cualitativa en cuestión.

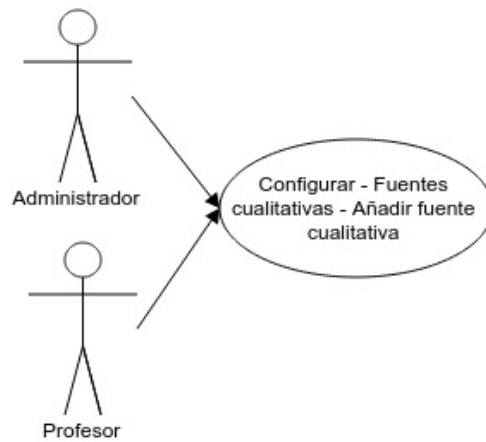


Figura 3.20: Caso de uso: Añadir fuente cualitativa

3. El sistema muestra un pop-up de confirmación.
4. El usuario confirma el borrado. La fuente cualitativa es borrada del sistema.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 4a. El usuario decide cancelar el borrado mediante el botón "Cancelar". En este caso el sistema no realiza ningún cambio en la base de datos y la fuente cualitativa sigue estando disponible.

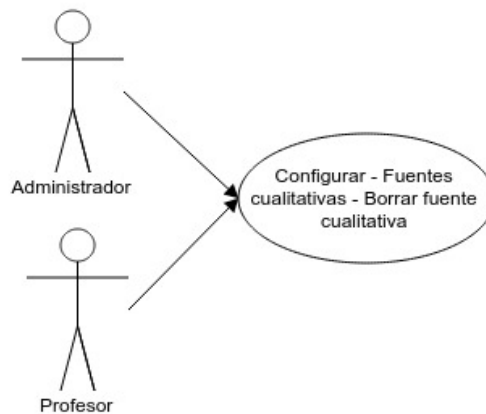


Figura 3.21: Caso de uso: Borrar fuente cualitativa

(Admin) Configurar - Usuarios - Ver lista de usuarios

Descripción: Ver listado de usuarios del sistema.

Actores: Administrador.

Resumen: Un administrador desea ver los usuarios registrados en el sistema.

Escenario principal:

1. El administrador desea ver una lista de los usuarios registrados en el sistema. Accede al apartado de configuración mediante el botón “Configuración”.
2. El administrador hace click en la pestaña “Usuarios”.
3. El sistema muestra una lista con todos los usuarios del sistema.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

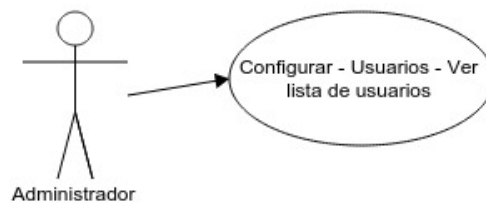


Figura 3.22: Caso de uso: Ver lista de usuarios

(Admin) Configurar - Usuarios - Añadir usuario

Descripción: Añadir un usuario al sistema.

Actores: Administrador.

Resumen: Un administrador desea registrar un nuevo usuario en el sistema.

Escenario principal:

1. El administrador desea registrar un nuevo usuario en el sistema. Accede al apartado de configuración mediante el botón “Configuración”.
2. El administrador hace click en la pestaña “Usuarios”.
3. El administrador pulsa el botón “Añadir usuario”.
4. El sistema muestra un pop-up con campos para rellenar con la información de usuario.
5. El administrador introduce la información del usuario y pulsa “Guardar”.
6. El sistema valida los datos y registra al usuario.

Escenarios alternativos:

- 0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 5a. El usuario pulsa “Cancelar”. En este caso se cierra el pop-up y no se registra ningún usuario.
- 6a. El sistema detecta que la información no está completa y muestra un mensaje de error. El usuario no se registra.
- 6b. El sistema detecta que el usuario ya existe y muestra un mensaje de error. El usuario no se registra.

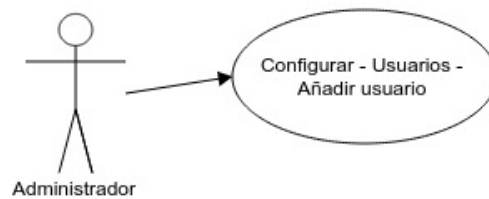


Figura 3.23: Caso de uso: Añadir usuario

(Admin) Configurar - Usuarios - Borrar usuario

Descripción: Borrar un usuario del sistema.

Actores: Administrador.

Resumen: Un administrador desea borrar un usuario del sistema.

Escenario principal:

1. El administrador desea borrar un usuario del sistema. Accede al apartado de configuración mediante el botón “Configuración”.
2. El administrador hace click en la pestaña “Usuarios”.
3. El administrador navega al usuario en cuestión y pulsa el botón de borrar.
4. El sistema muestra un pop-up para confirmar el borrado del usuario.
5. El administrador confirma el borrado pulsando “Borrar”.
6. El sistema elimina al usuario.

Escenarios alternativos:

- 0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.
- 5a. El usuario pulsa “Cancelar”. En este caso se cierra el pop-up y no se borra ningún usuario.

(Admin) Configurar - Añadir función de calidad

Descripción: Añadir nueva función de calidad.

Actores: Administrador.

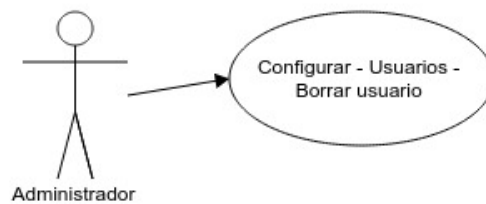


Figura 3.24: Caso de uso: Borrar usuario

Resumen: Un administrador desea añadir una nueva función de calidad para personalizar los cálculos del un usuario.

Escenario principal:

1. El administrador desea añadir una nueva función de calidad y abre el archivo de configuración de back-end de CleverFigures.
2. El administrador añade la nueva función en el listado de funciones disponibles.
3. El administrador guarda el archivo.
4. El sistema habilita la función añadida para el resto de usuarios a partir de dicho momento.

Escenarios alternativos:

0. El usuario puede cerrar el archivo en cualquier momento sin guardar y todos los cambios serán revertidos.

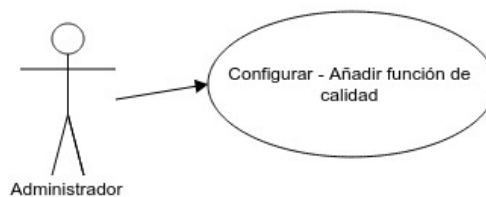


Figura 3.25: Caso de uso: Añadir función de calidad

(Admin) Configurar - Añadir datos de contacto

Descripción: Añadir información de contacto.

Actores: Administrador.

Resumen: Un administrador desea añadir información de contacto.

Escenario principal:

1. El administrador desea añadir información de contacto y abre el archivo de configuración de front-end de CleverFigures.

2. El administrador añade los nuevos datos de contacto en las variables creadas para ello.
3. El administrador guarda el archivo.
4. El sistema publica dichos datos en el apartado de contacto de la aplicación para hacerlos disponibles a los usuarios.

Escenarios alternativos:

0. El usuario puede cerrar el archivo en cualquier momento sin guardar y todos los cambios serán revertidos.

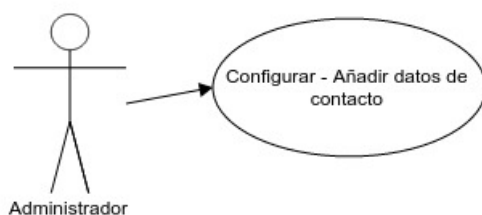


Figura 3.26: Caso de uso: Añadir datos de contacto

Contacto - Ver información de contacto

Descripción: Ver información de contacto.

Actores: Profesor, Administrador.

Resumen: Un usuario desea ver la información de contacto.

Escenario principal:

1. El usuarios desea ver la información de contacto y abre el apartado de “Contacto” de la aplicación.
2. El sistema muestra los datos de contacto con el administrador y el soporte técnico.

Escenarios alternativos:

0. El usuario puede cerrar la ventana del navegador y abandonar la aplicación en cualquier momento.

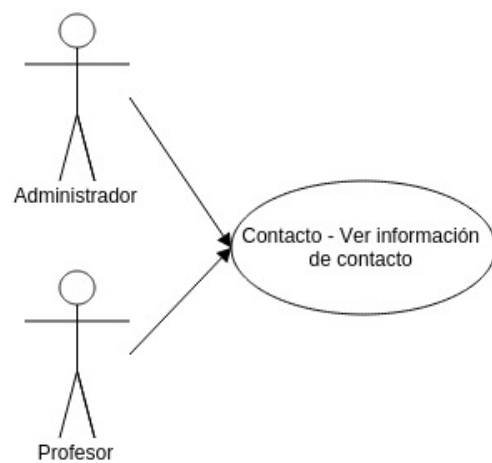


Figura 3.27: Caso de uso: Ver información de contacto

3.4.3. Modelo de comportamiento

A continuación podemos ver los diagramas de secuencia para cada caso de uso descrito anteriormente.

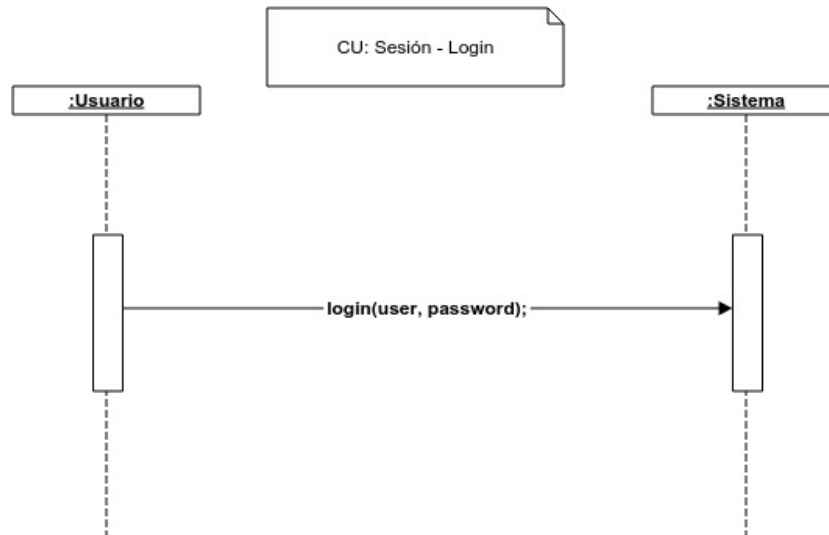


Figura 3.28: Modelo de comportamiento: Login de usuario

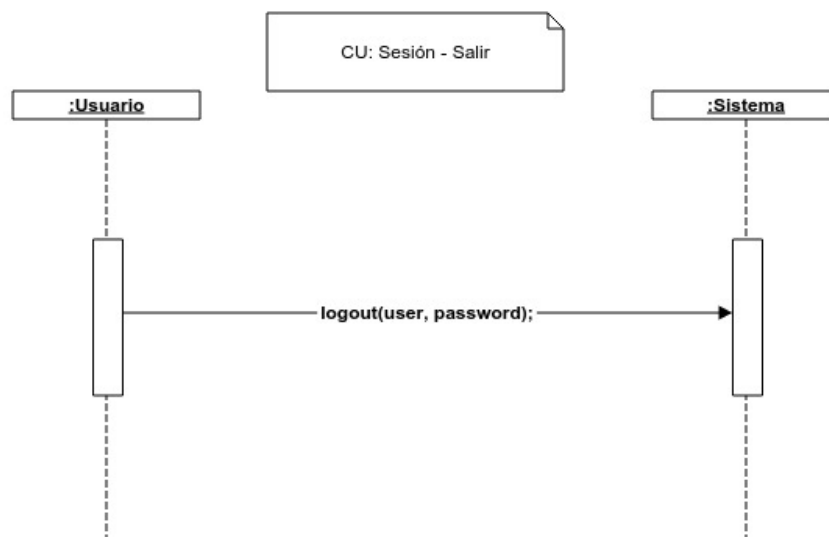


Figura 3.29: Modelo de comportamiento: Logout de usuario

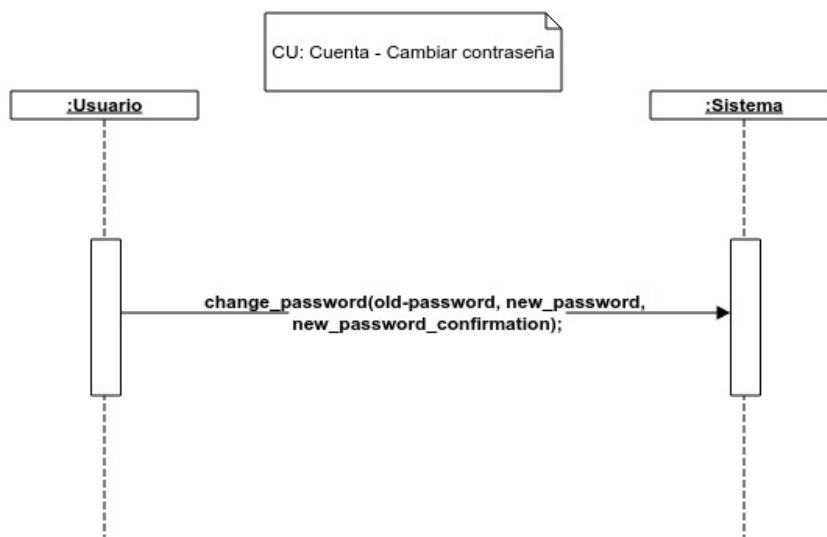


Figura 3.30: Modelo de comportamiento: Cambiar contraseña

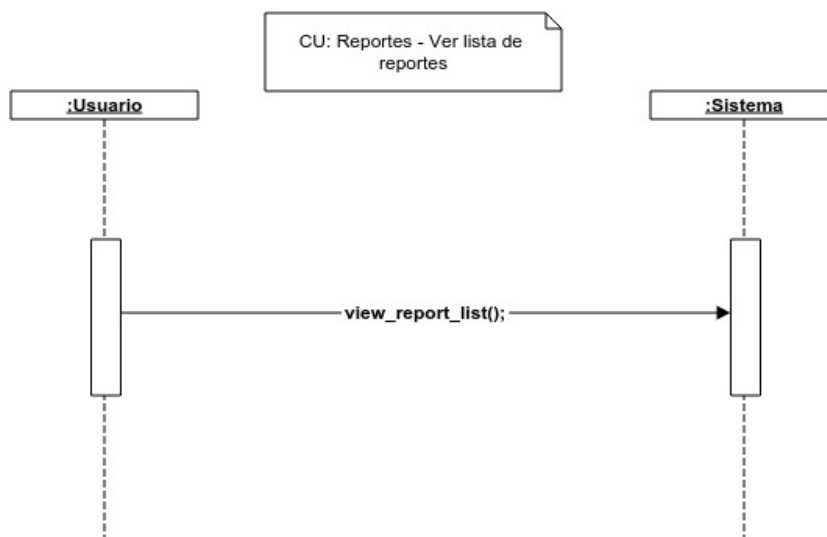


Figura 3.31: Modelo de comportamiento: Ver listado de reportes

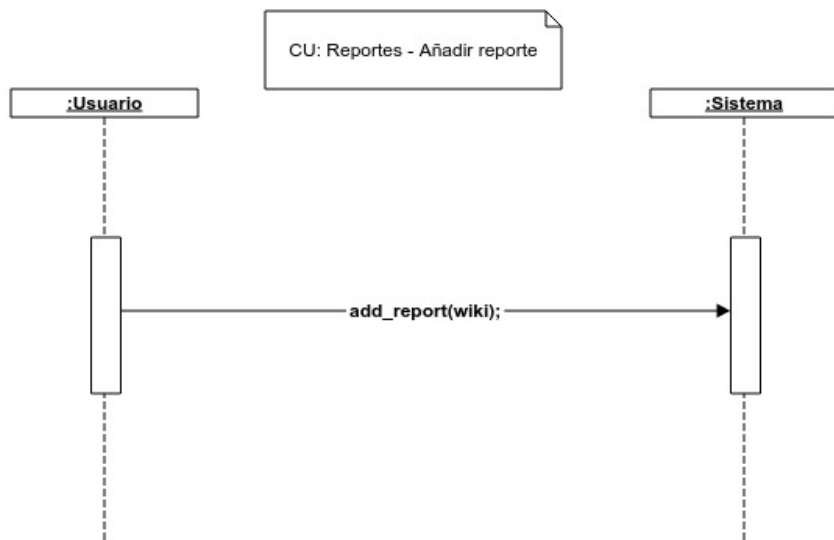


Figura 3.32: Modelo de comportamiento: Añadir reporte

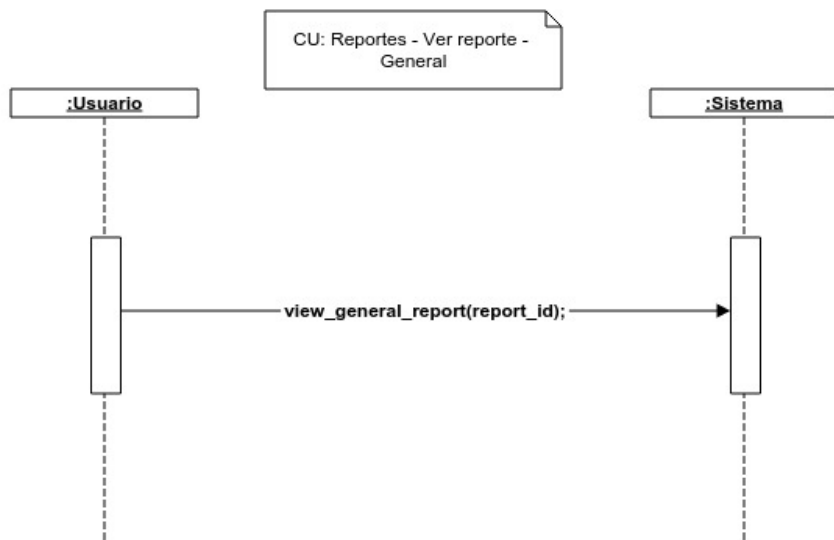


Figura 3.33: Modelo de comportamiento: Ver reporte general

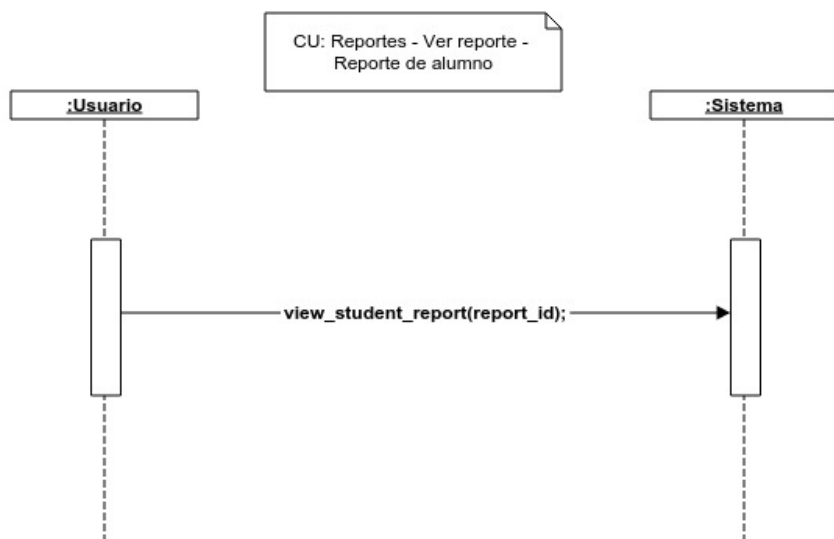


Figura 3.34: Modelo de comportamiento: Ver reporte de alumno

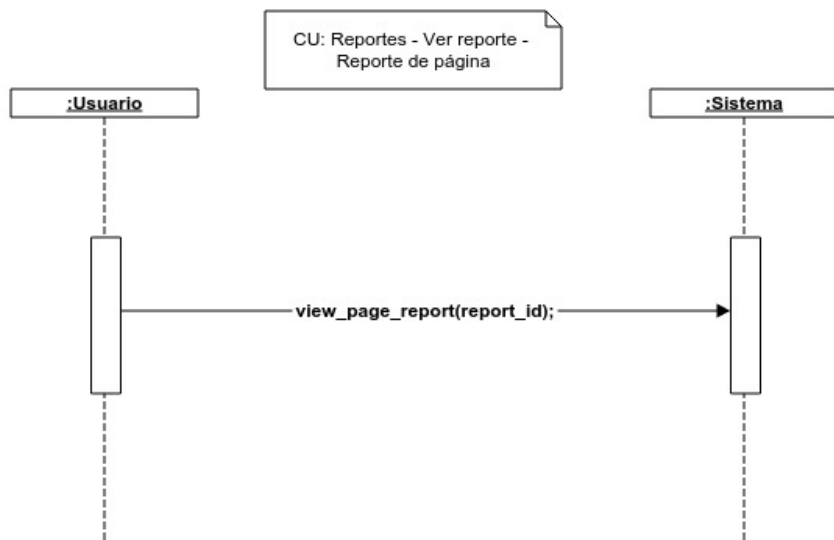


Figura 3.35: Modelo de comportamiento: Ver listado de página

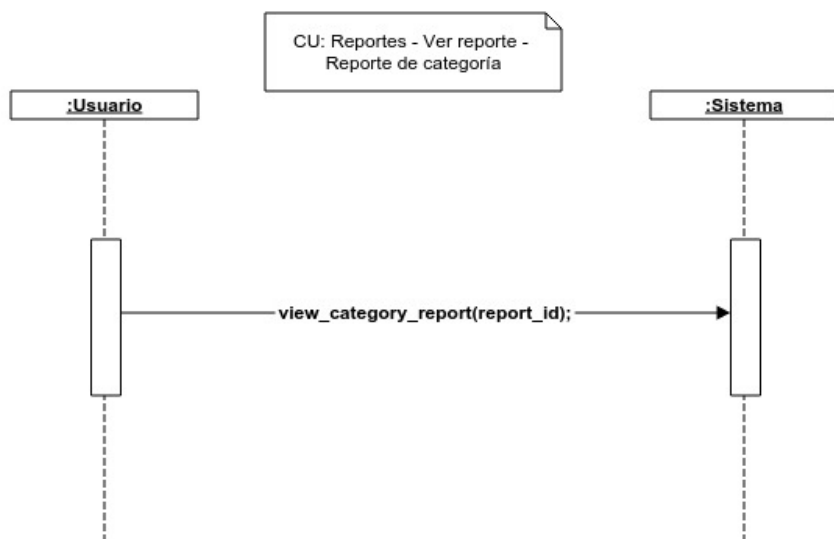


Figura 3.36: Modelo de comportamiento: Ver reporte de categoría

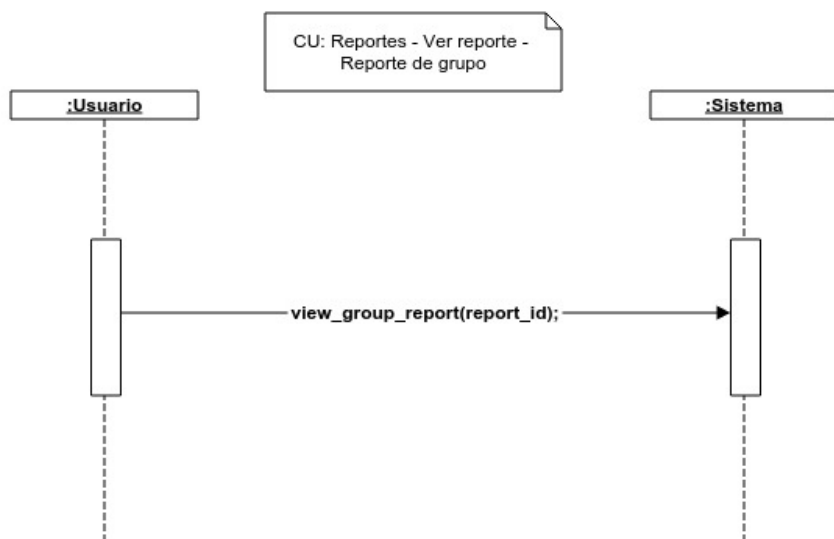


Figura 3.37: Modelo de comportamiento: Ver reporte de grupo

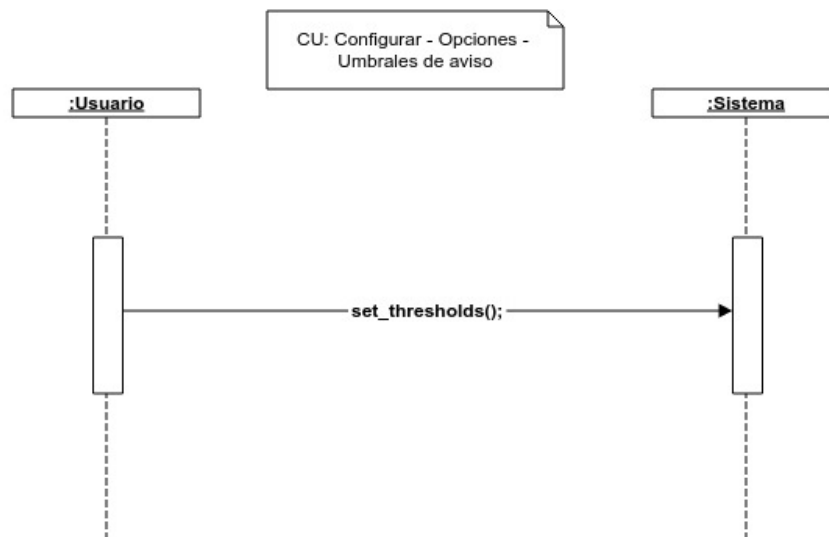


Figura 3.38: Modelo de comportamiento: Umbrales de aviso

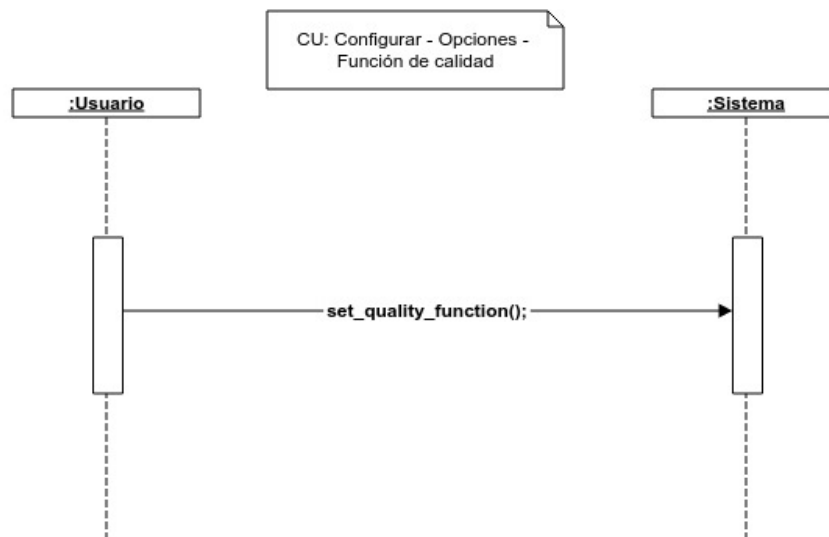


Figura 3.39: Modelo de comportamiento: Función de calidad

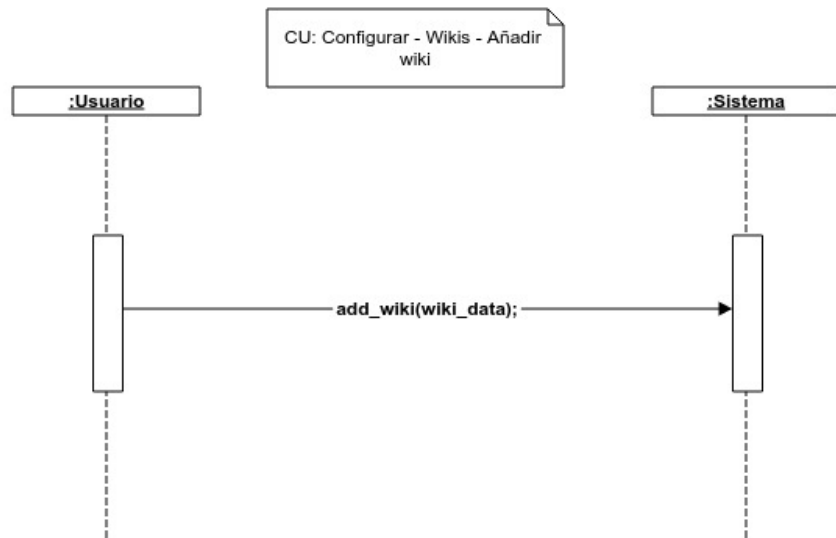


Figura 3.40: Modelo de comportamiento: Añadir wiki

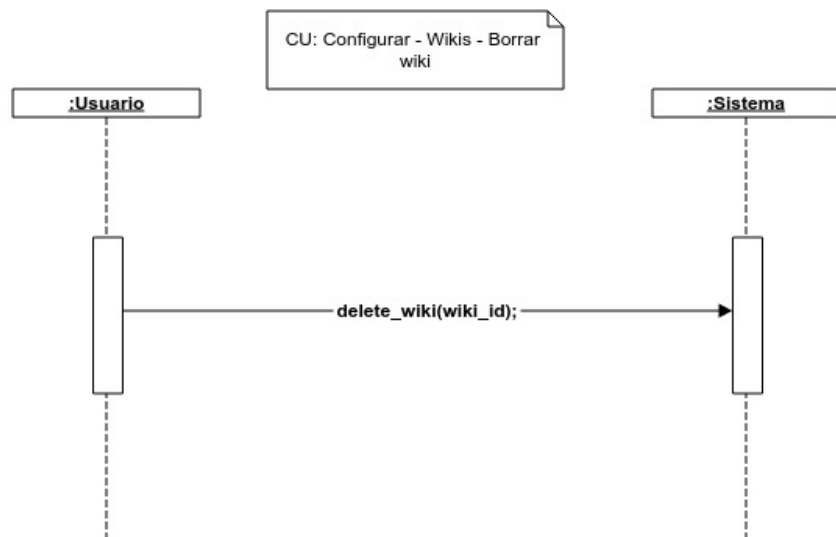


Figura 3.41: Modelo de comportamiento: Borrar wiki

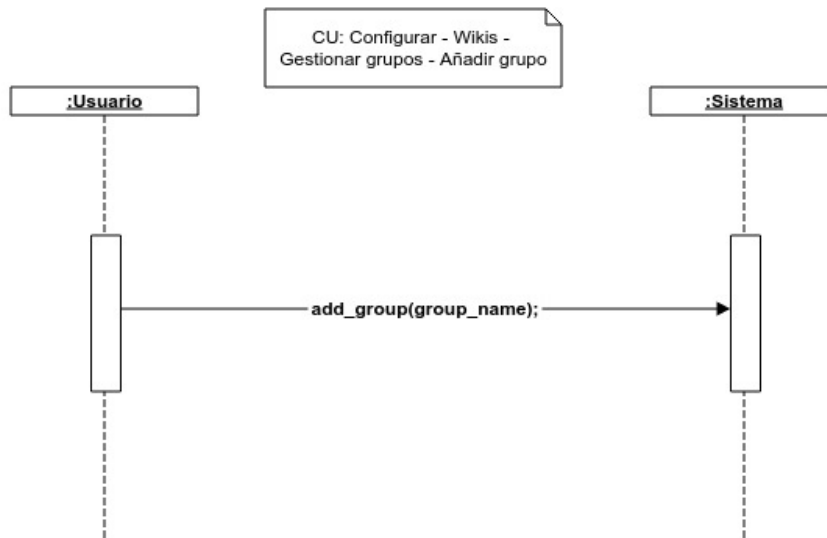


Figura 3.42: Modelo de comportamiento: Añadir grupo

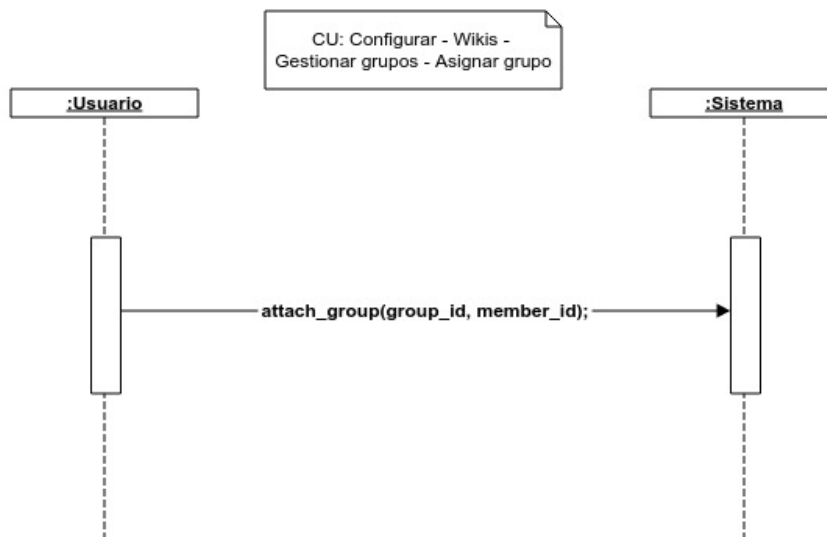


Figura 3.43: Modelo de comportamiento: Asignar grupo

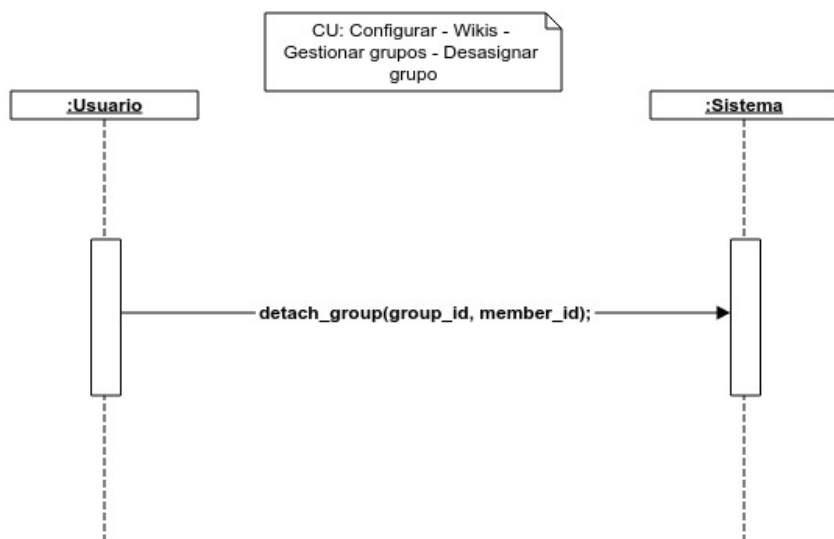


Figura 3.44: Modelo de comportamiento: Desasignar grupo

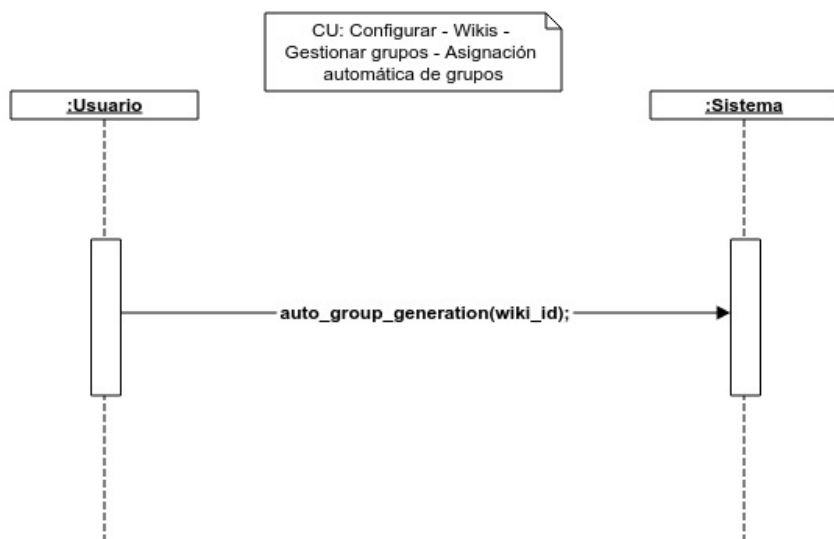


Figura 3.45: Modelo de comportamiento: Asignación automática de grupos

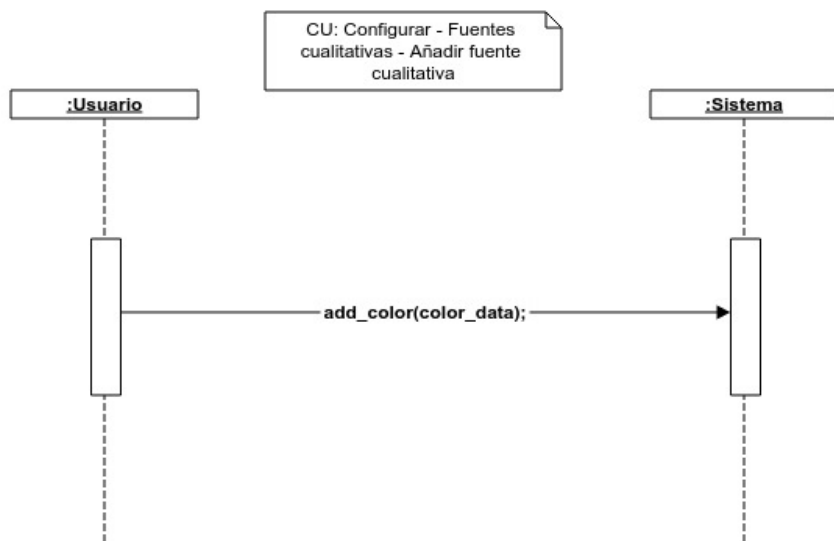


Figura 3.46: Modelo de comportamiento: Añadir fuente cualitativa

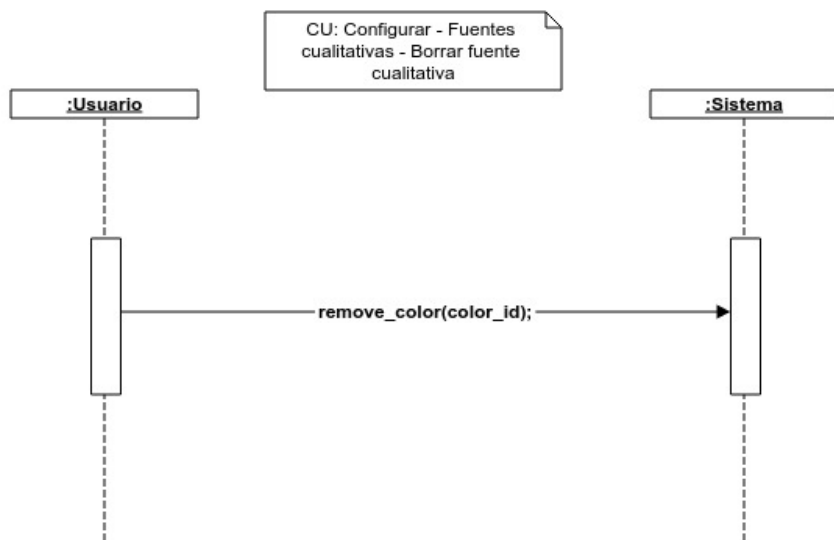


Figura 3.47: Modelo de comportamiento: Borrar fuente cualitativa

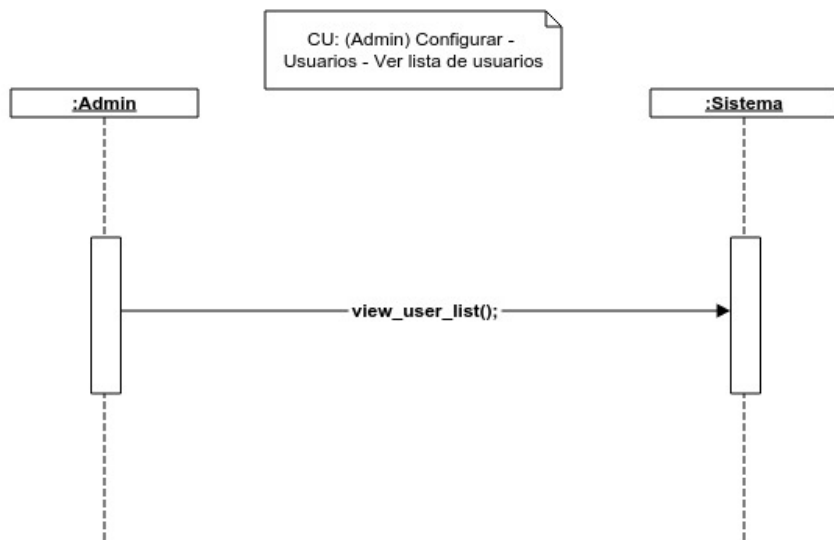


Figura 3.48: Modelo de comportamiento: Ver lista de usuarios

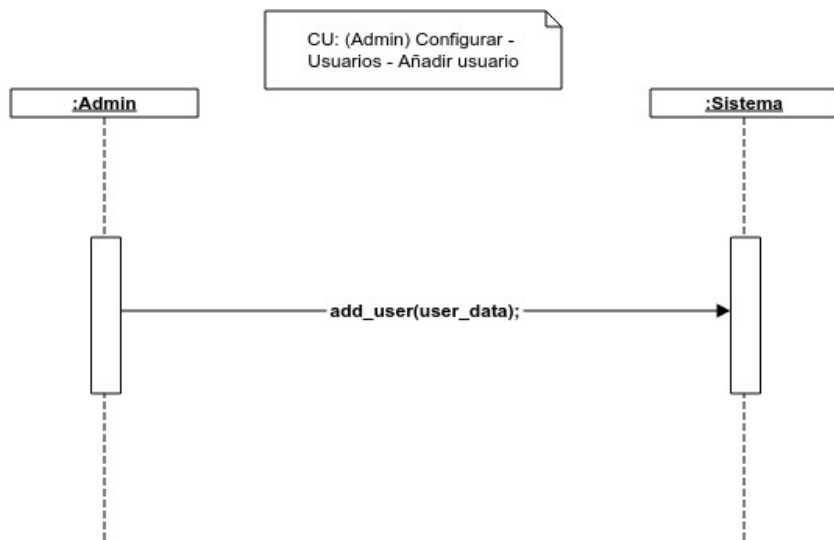


Figura 3.49: Modelo de comportamiento: Añadir usuario

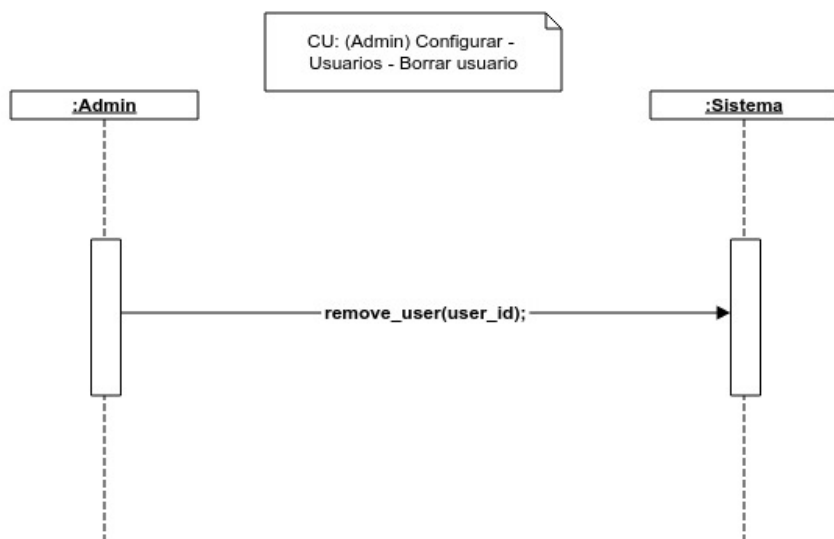


Figura 3.50: Modelo de comportamiento: Borrar usuario

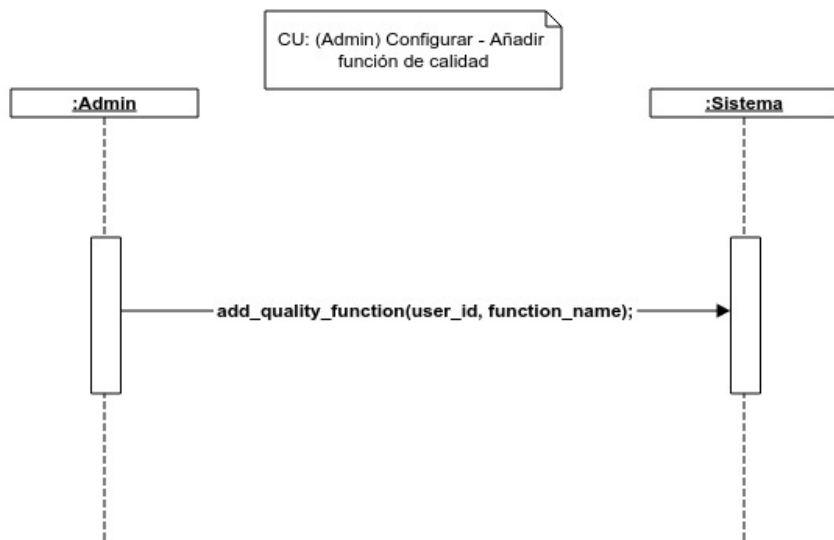


Figura 3.51: Modelo de comportamiento: Añadir función de calidad

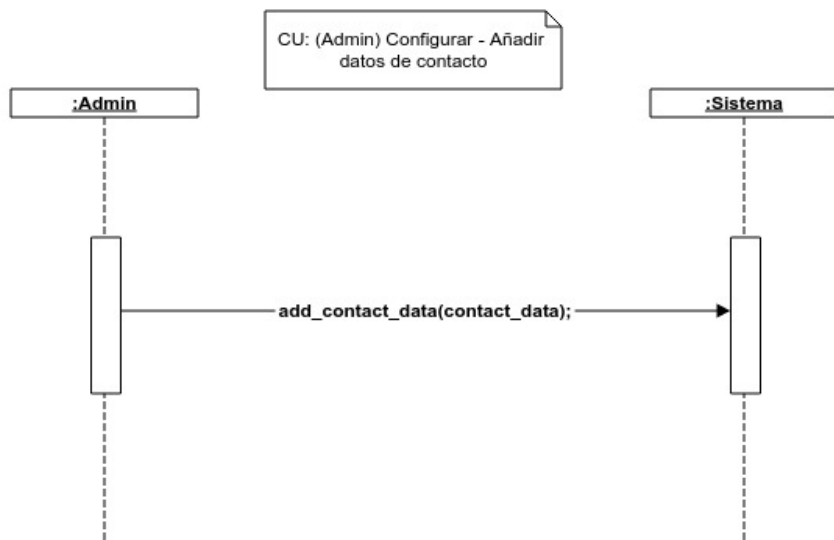


Figura 3.52: Modelo de comportamiento: Añadir datos de contacto

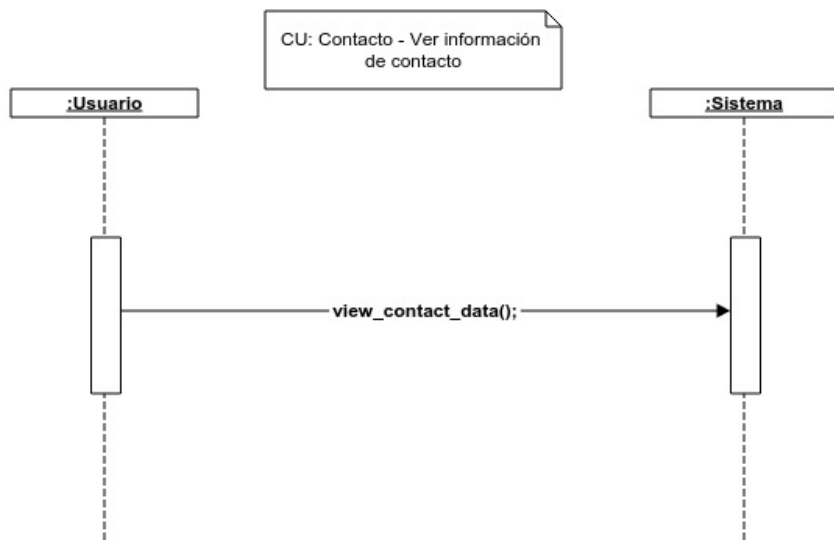


Figura 3.53: Modelo de comportamiento: Ver información de contacto

3.4.4. Contratos de operaciones del sistema

Operación: login(user, pass)

Responsabilidades: Iniciar la sesión del usuario en el sistema.

Referencias cruzadas: Caso de uso “Sesión - Login”.

Pre-condiciones:

- Existe un usuario con el nombre de usuario dado.
- El usuario registrado tiene como contraseña la recibida por la función.

Post-condiciones:

- El usuario está logado en el sistema y tiene acceso a las funcionalidades de la aplicación.

Operación: logout(user, pass)

Responsabilidades: Cierra la sesión del usuario en el sistema.

Referencias cruzadas: Caso de uso “Sesión - Salir”.

Pre-condiciones:

- Existe un usuario logado en el sistema.

Post-condiciones:

- El usuario queda fuera del sistema y es redirigido a la pantalla de login.

Operación: add_contact_data(contact_data)

Responsabilidades: Añade los datos de contacto de la aplicación.

Referencias cruzadas: Caso de uso “(Admin) Configurar - Añadir datos de contacto”.

Pre-condiciones:

- El usuario está logado y es administrador.
- contact_data no es nulo.

Post-condiciones:

- Los datos de contacto de la aplicación quedan registrados y son visibles a todos los usuarios.

Operación: add_quality_function(user_id, function_name)

Responsabilidades: Añade una nueva función de cálculo de calidad al sistema.

Referencias cruzadas: Caso de uso “(Admin) Configurar - Añadir función de calidad”.

Pre-condiciones:

- El usuario está logado y es administrador.
- user_id es un id de usuario válido.
- function_name no es nulo.

Post-condiciones:

- La función queda registrada en el sistema para usuario el usuario dado.

Operación: add_user(user_data)

Responsabilidades: Añade un nuevo usuario al sistema.

Referencias cruzadas: Caso de uso “(Admin) Configurar - Usuarios - Añadir usuario”.

Pre-condiciones:

- El usuario está logado y es administrador.
- user_data no es nulo.

Post-condiciones:

- El usuario queda registrado en el sistema y puede acceder al mismo a partir de ese instante.

Operación: remove_user(user_id)

Responsabilidades: Elimina un nuevo usuario del sistema.

Referencias cruzadas: Caso de uso “(Admin) Configurar - Usuarios - Borrar usuario”.

Pre-condiciones:

- El usuario está logado y es administrador.
- user_id no es nulo.

Post-condiciones:

- El usuario es eliminado del sistema y no puede acceder al mismo a partir de ese instante.

Operación: view_user_list()

Responsabilidades: Lista los usuarios del sistema.

Referencias cruzadas: Caso de uso “(Admin) Configurar - Usuarios - Ver lista usuarios”.

Pre-condiciones:

- El usuario está logado y es administrador.

Post-condiciones:

- El sistema muestra una lista de los usuarios registrados.

Operación: add_color(color_data)

Responsabilidades: Añade una fuente cualitativa al sistema.

Referencias cruzadas: Caso de uso “Configurar - Fuentes cualitativas - Añadir fuente cualitativa”.

Pre-condiciones:

- El usuario está logado.
- color_data no es nulo.

Post-condiciones:

- La fuente es añadida al sistema y está disponible para su uso a partir de ese instante.

Operación: remove_color(color_id)

Responsabilidades: Elimina una fuente cualitativa del sistema.

Referencias cruzadas: Caso de uso “Configurar - Fuentes cualitativas - Borrar fuente cualitativa”.

Pre-condiciones:

- El usuario está logado.
- color_id no es nulo.

Post-condiciones:

- La fuente es eliminada del sistema y no está disponible para su uso a partir de ese instante.

Operación: set_quality_function()

Responsabilidades: Establece la función de calidad a usar en los reportes.

Referencias cruzadas: Caso de uso “Configurar - Opciones - Función de calidad”.

Pre-condiciones:

- El usuario está logado.

Post-condiciones:

- Configura la función de calidad previamente añadida por un administrador.

Operación: set_thresholds()

Responsabilidades: Establece los umbrales de aviso a usar en los reportes.

Referencias cruzadas: Caso de uso “Configurar - Opciones - Umbrales”.

Pre-condiciones:

- El usuario está logado.

Post-condiciones:

- Establece los umbrales a los valores dados por el usuario, y son tenidos en cuenta a partir de ese instante a la hora de generar los reportes.

Operación: add_wiki(wiki_data)

Responsabilidades: Añade una wiki al sistema.

Referencias cruzadas: Caso de uso “Configurar - Fuentes cualitativas - Añadir wiki”.

Pre-condiciones:

- El usuario está logado.
- wiki_data no es nulo.

Post-condiciones:

- La wiki es añadida al sistema y está disponible para su uso a partir de ese instante.

Operación: delete_wiki(wiki_id)

Responsabilidades: Elimina una wiki del sistema.

Referencias cruzadas: Caso de uso “Configurar - Fuentes cualitativas - Borrar wiki”.

Pre-condiciones:

- El usuario está logado.
- wiki_id no es nulo.

Post-condiciones:

- La wiki es eliminada del sistema y no está disponible para su uso a partir de ese instante.

Operación: attach_group(group_id, member_id)

Responsabilidades: Asigna un alumno a un grupo.

Referencias cruzadas: Caso de uso “Configurar - Wikis - Gestionar grupos - Asignar grupo”.

Pre-condiciones:

- El usuario está logado.
- group_id no es nulo.
- member_id no es nulo.

Post-condiciones:

- El alumno member_id es asignado al grupo group_id.

Operación: add_group(group_name)

Responsabilidades: Crea un nuevo grupo.

Referencias cruzadas: Caso de uso “Configurar - Wikis - Gestionar grupos - Crear grupo”.

Pre-condiciones:

- El usuario está logado.
- group_name no es nulo.

Post-condiciones:

- El grupo group_name es creado y está disponible para asignar a partir de ese instante.

Operación: auto_group_generation(wiki_id)

Responsabilidades: Crea grupos de alumnos automáticamente a partir de su trabajo en las páginas.

Referencias cruzadas: Caso de uso “Configurar - Wikis - Gestionar grupos - Asignación automática de grupos”.

Pre-condiciones:

- El usuario está logado.
- wiki_id no es nulo.

Post-condiciones:

- Se crean todos los grupos actuales para la wiki wiki_id.

Operación: detach_group(group_id, member_id)

Responsabilidades: Desasigna un grupo de un alumno.

Referencias cruzadas: Caso de uso “Configurar - Wikis - Gestionar grupos - Desasignar grupo”.

Pre-condiciones:

- El usuario está logado.
- group_id no es nulo.
- member_id no es nulo.

Post-condiciones:

- Se desasigna el grupo group_id del alumno member_id.

Operación: view_contact_data()

Responsabilidades: Ver datos de contacto del sistema.

Referencias cruzadas: Caso de uso “Contacto - Ver información de contacto”.

Pre-condiciones:

- El usuario está logado.

Post-condiciones:

- El sistema muestra los datos de contacto del sistema.

Operación: change_password(old_password, new_password, new_password_confirmation)

Responsabilidades: Cambiar contraseña del usuario.

Referencias cruzadas: Caso de uso “Cuenta - Cambiar contraseña”.

Pre-condiciones:

- El usuario está logado.
- old_password no es nula y es igual a la contraseña actual del usuario.
- new_password no es nula.
- new_password confirmation no es nula y es igual a new_password.

Post-condiciones:

- El sistema cambia la contraseña del usuario a new_password. El usuario deberá usarla para acceder a partir de ese instante.

Operación: add_report(wiki)

Responsabilidades: Añade un reporte de una wiki dada.

Referencias cruzadas: Caso de uso “Reportes - Añadir reporte”.

Pre-condiciones:

- El usuario está logado.
- wiki no es nula.

Post-condiciones:

- El reporte de la wiki queda guardado y puede ser visualizado a partir de ese instante.

Operación: view_report_list()

Responsabilidades: Lista los reportes existentes.

Referencias cruzadas: Caso de uso “Reportes - Ver lista de reportes”.

Pre-condiciones:

- El usuario está logado.

Post-condiciones:

- El sistema muestra una lista de los reportes creados por el usuario.

Operación: view_general_report(report_id)

Responsabilidades: Muestra el reporte general dado.

Referencias cruzadas: Caso de uso “Reportes - Ver reporte - General”.

Pre-condiciones:

- El usuario está logado.
- report_id no es nulo.

Post-condiciones:

- El sistema muestra el reporte general report_id al usuario.

Operación: view_student_report(report_id)

Responsabilidades: Muestra el reporte de alumno dado.

Referencias cruzadas: Caso de uso “Reportes - Ver reporte - Reporte de alumno”.

Pre-condiciones:

- El usuario está logado.
- report_id no es nulo.

Post-condiciones:

- El sistema muestra el reporte general de alumno report_id al usuario.

Operación: view_category_report(report_id)

Responsabilidades: Muestra el reporte de categoría dado.

Referencias cruzadas: Caso de uso “Reportes - Ver reporte - Reporte de categoría”.

Pre-condiciones:

- El usuario está logado.
- report_id no es nulo.

Post-condiciones:

El sistema muestra el reporte de categoría report_id al usuario.

Operación: view_group_report(report_id)

Responsabilidades: Muestra el reporte de grupo dado.

Referencias cruzadas: Caso de uso “Reportes - Ver reporte - Reporte de grupo”.

Pre-condiciones:

- El usuario está logado.
- report_id no es nulo.

Post-condiciones:

- El sistema muestra el reporte de grupo report_id al usuario.

Operación: view_page_report(report_id)

Responsabilidades: Muestra el reporte de página dado.

Referencias cruzadas: Caso de uso “Reportes - Ver reporte - Reporte de página”.

Pre-condiciones:

- El usuario está logado.
- report_id no es nulo.

Post-condiciones:

- El sistema muestra el reporte de página report_id al usuario.

3.5. Especificación de requisitos

A continuación se describen los distintos requisitos del sistema, clasificados en requisitos gráficos (de la interfaz de usuario y los reportes), requisitos funcionales (relacionados con el funcionamiento y la lógica de negocio) y requisitos de rendimiento (uso de recursos del sistema).

3.5.1. Requisitos gráficos

Como en cualquier aplicación web, la interfaz de usuario debe estar orientada a la usabilidad. En nuestro caso, lo importante de la aplicación no es la interfaz en sí, sino los reportes, por lo que se necesita una interfaz simplista, fácil de usar, e intuitiva. Aunque se pretende no molestar al usuario con una interfaz detallista, se requiere que la misma tenga un aspecto moderno y acorde con la web actual.

En cuanto a los reportes, deben seguir la misma línea que la interfaz en general, pero enfocándose en el buen orden y clasificación de los gráficos.

En resumen:

- Diseño simplista y moderno.
- Acorde con los estándares de usabilidad actuales.
- Interfaz intuitiva y fácil de usar.
- Reportes clasificados correctamente.

3.5.2. Requisitos funcionales

Login

- El usuario debe poder acceder a la aplicación usando un nombre de usuario y contraseña. Esta contraseña será almacenada de forma segura y no será recuperable. Deberá ser cambiada en caso de pérdida.
- Así como puede entrar en el sistema, el usuario debe poder cerrar su sesión de forma rápida y directa.

Configuración

- La configuración de los distintos parámetros estará dividida en varias pestañas: opciones, wikis, fuentes cualitativas y usuarios.
- En la pestaña de opciones, el usuario dispondrá de un listado de umbrales que podrá configurar a su gusto. Estos umbrales crearán avisos en los informes, para alertar al profesor cuando alguno de los parámetros no llegue al mínimo, o supere el máximo de alguno de ellos.
- En la misma pestaña, se podrá configurar qué función de cálculo de la calidad se quiere usar. Estas funciones deben ser aprobadas por el administrador previamente, para evitar agujeros de seguridad.
- En la pestaña de wikis, el usuario puede gestionar sus conexiones a wikis: añadir, listar y borrar wikis. Para cada wiki, además, podrá gestionar grupos de alumnos: crear grupos y asignar alumnos a dichos grupos. Para poder añadir una nueva wiki, el usuario deberá disponer de los datos de conexión de su base de datos.
- En la pestaña de fuentes cualitativas, el usuario puede gestionar sus fuentes cualitativas: añadir, listar y borrar fuentes. Para poder añadir una nueva fuente, el usuario deberá disponer de los datos de conexión de su base de datos.
- En la pestaña de usuarios, el usuario (en este caso solo el administrador) podrá gestionar los usuarios del sistema. Se podrán registrar nuevos usuarios y eliminar los existentes.

Reportes

- El usuario debe poder crear, borrar, listar y visualizar reportes.
- Desde el listado de reportes se podrán ver y borrar los reportes. Al visualizar un reporte, se debe poder navegar por el mismo de forma intuitiva.
- Los avisos personalizados estarán integrados en cada reporte.
- El usuario dispondrá de un botón mediante el cual podrá crear nuevos reportes, diciendo qué wiki y fuente cualitativa quiere usar.

Contacto

- El usuario debe disponer de una sección de contacto con información de contacto clara a la que acudir cuando tenga dudas o problemas técnicos. Esta información debe poder ser editable por el administrador de la aplicación.

3.5.3. Requisitos de rendimiento

Nos encontramos ante una aplicación en la que las respuestas en tiempo real no son un requisito. Debido a la naturaleza de la misma, no es necesario que las respuestas del sistema se den en un periodo de tiempo crítico. Sin embargo, para mejorar la experiencia del usuario, se establecen las siguientes restricciones referentes a los tiempos de respuesta de las llamadas a la API del servidor. Los tiempos son medidos usando la consola de depuración de distintos navegadores web (Google Chrome, Mozilla Firefox y Opera), con los equipos cliente y servidor en una red local cableada de extensión menor a 100 metros, y con el cliente con una carga inferior a 0.5 en el último minuto.

- En todas las acciones que se realicen en la aplicación, el tiempo de respuesta no debe ser mayor a 200 milisegundos.
- El caso de la visualización de reportes es una excepción a la regla anterior. Debido al gran volumen de datos que manejan y la gran carga que supone para el lado del cliente generar los gráficos, el tiempo máximo de carga de la página se extiende a 1 segundo. En cualquier caso, ninguna de las sub-llamadas que se hacen para generar los gráficos, debe superar los 200 milisegundos.

Otro aspecto a tener en cuenta es la carga del sistema. En la medida de lo posible, se debe reducir el uso del procesador, tanto del servidor como del cliente. Los algoritmos usados no deben superar el orden N^2 en el peor caso [13, 14, 15].

En cuanto al uso del sistema de ficheros, la aplicación no debe crear, en ningún caso, ficheros temporales, lo que ralentizaría los procesos. Así mismo, los accesos a la base de datos deben reducirse al máximo posible. Se deberán realizar consultas concretas, que aprovechen el carácter relacional de la base de datos y los índices optimizados.

Capítulo 4

Diseño

En esta sección se describe cómo estará diseñado el sistema descrito anteriormente. La mayoría de las decisiones relacionadas con el diseño fueron tomadas por el alumno desarrollador del proyecto teniendo siempre en mente el rendimiento y la escalabilidad de la aplicación resultante. Se pretendía que el resultado, además de cumplir su objetivo con el personal docente, facilitara el trabajo de los posibles futuros desarrolladores que pretendieran extender las funcionalidades del sistema. Cabe destacar que, debido a la interrupción del desarrollo por razones personales (reflejado en el diagrama de Gantt adjunto), el proyecto sufrió un rediseño y reimplementación, derivados de mi experiencia profesional adquirida durante esta interrupción.

4.1. Arquitectura del Sistema

En el proceso de arquitectura se decidió que la mejor opción era disponer de una API en lado del servidor y dividida en módulos, que consumiera datos tanto de las wikis como de Assess-MediaWiki. Esta API sería la encargada de obtener los datos necesarios para la generación de los informes desde las bases de datos y de transmitirlos mediante llamadas HTTP a la interfaz de la aplicación. Así mismo, se incluirían también en la API las funciones de generación y almacenamiento del historial de informes y del resto de configuraciones en general, siendo posible acceder a todas las funcionalidades del sistema de forma programática. Para cumplir el objetivo era necesario también implementar una interfaz de usuario que facilitara la lectura de los informes. Esta interfaz estaría, como dijimos anteriormente, disponible a través del navegador web. Su implementación debía tener entre sus principales objetivos el ofrecer un buen rendimiento, por lo que se decidió que consumiría de la API mediante llamadas asíncronas. El resto de la implementación debía tener esto en cuenta para no generar condiciones de carrera e inconsistencia de datos derivadas de estas.

4.2. Modelo de datos

Para poder almacenar las configuraciones pertinentes, los grupos de alumnos y demás datos inherentes a CleverFigures, el sistema debía disponer de una base de datos propia. Esta base de datos debía diseñarse de forma que facilitara y optimizara las consultas relacionales, sobre todo entre wikis, personal docente e informes.

Como podemos apreciar en el esquema, la base de datos principal de CleverFigures sirve para alojar lo usuarios del sistema, las wikis, las fuentes cualitativas, los parámetros y los análisis

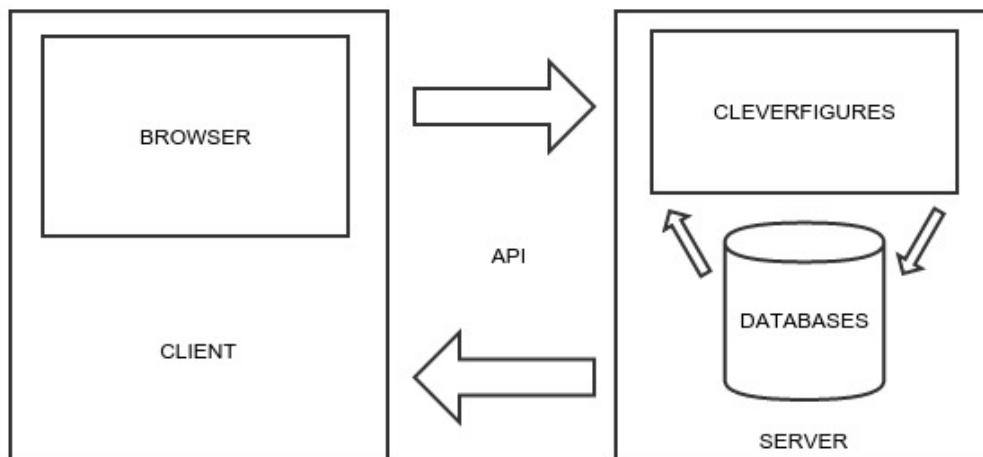


Figura 4.1: Arquitectura del sistema

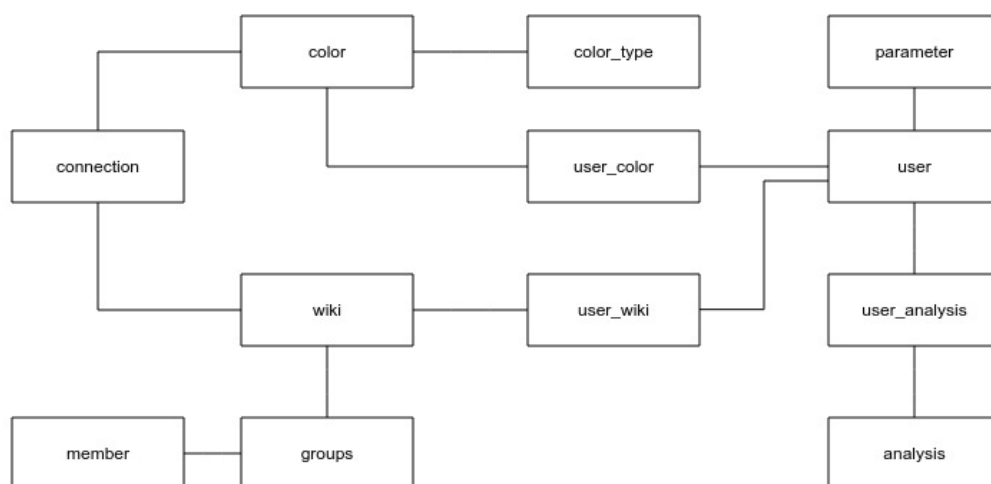


Figura 4.2: Modelo de datos

(reportes). A parte de las tablas específicas para dichas entidades, podemos ver también las que sirven para relacionarlas. A continuación podemos ver más en detalle la definición de cada tabla:

También se aportan los esquemas de los modelos de datos correspondientes a MediaWiki y AssessMediaWiki, que han sido utilizados para obtener información de dichos sistemas.

En concreto, de MediaWiki usamos las tablas user, page, revision, category, categorylinks, image y oldimage. De ellas obtenemos toda la información numérica sobre el trabajo de los alumnos, la cual procesamos más tarde para obtener un conocimiento más interesante.[12]

De AssessMediaWiki, usamos las tablas entregable, evaluaciones, evaluaciones_entregables y replies. De ellas obtenemos la información cualitativa necesaria para enriquecer los reportes de CleverFigures, resultado de procesar los datos numéricos obtenidos previamente de la wiki.[11]

Field	Type	Null	Key	Default	Extra
analysis_id	int(11)	NO	MUL	NULL	auto_increment
analysis_date	int(50)	NO	MUL	NULL	
analysis_wiki	int(11)	NO	MUL	NULL	
analysis_color	int(11)	YES		NULL	

Cuadro 4.1: Tabla analysis (reportes)

Field	Type	Null	Key	Default	Extra
color_id	int(11)	NO	MUL	NULL	auto_increment
color_name	varchar(50)	NO		NULL	
color_connection	int(11)	NO		NULL	
color_type	int(11)	NO		NULL	
color_wiki	int(11)	NO		NULL	

Cuadro 4.2: Tabla color (fuentes cualitativas)

Field	Type	Null	Key	Default	Extra
ct_id	int(11)	NO	PRI	NULL	auto_increment
ct_name	varchar(50)	NO		NULL	

Cuadro 4.3: Tabla color_type (relación de fuentes cualitativas con tipos de fuentes cualitativas)

Field	Type	Null	Key	Default	Extra
connection_id	int(11)	NO	MUL	NULL	auto_increment
connection_name	varchar(256)	NO		NULL	
connection_server	varchar(256)	NO		NULL	
connection_user	varchar(256)	NO		NULL	
connection_password	varchar(256)	NO		NULL	

Cuadro 4.4: Tabla connection (credenciales de conexión para bases de datos de wikis y fuentes cualitativas)

Field	Type	Null	Key	Default	Extra
group_id	int(11)	NO	PRI	NULL	auto_increment
group_name	varchar(50)	NO	UNI	NULL	
group_wiki	int(11)	NO	MUL	NULL	

Cuadro 4.5: Tabla groups (grupos de alumnos)

Field	Type	Null	Key	Default	Extra
member_id	int(11)	NO	PRI	NULL	auto_increment
member_name	varchar(20)	NO		NULL	
member_group	int(11)	NO	MUL	NULL	
member_wiki_id	int(11)	NO		NULL	

Cuadro 4.6: Tabla member (miembros de los grupos)

Field	Type	Null	Key	Default	Extra
parameter_id	int(11)	NO	PRI	NULL	auto_increment
parparameter_name	varchar(128)	NO	UNI	NULL	
parparameter_value	varchar(512)	NO		NULL	
parparameter_user	int(11)	YES	MUL	NULL	

Cuadro 4.7: Tabla parameter (opciones de configuración de usuarios)

Field	Type	Null	Key	Default	Extra
user_id	int(11)	NO	PRI	NULL	auto_increment
user_username	varchar(20)	NO	UNI	NULL	
user_password	varchar(50)	NO		NULL	
user_last_session	int(50)	NO		NULL	
user_realname	varchar(128)	NO		NULL	
user_email	varchar(128)	NO		NULL	
user_language	varchar(10)	NO		NULL	
user_is_admin	tinyint(1)	NO		NULL	

Cuadro 4.8: Tabla user (usuarios del sistema)

Field	Type	Null	Key	Default	Extra
ua_user	int(11)	NO	PRI	NULL	
ua_analysis	int(11)	NO	PRI	NULL	

Cuadro 4.9: Tabla user_analysis (relaciones de usuarios con reportes)

Field	Type	Null	Key	Default	Extra
uc_user	int(11)	NO	PRI	NULL	
uc_color	int(11)	NO	PRI	NULL	

Cuadro 4.10: Tabla user_color (relaciones de usuarios con fuentes cualitativas)

Field	Type	Null	Key	Default	Extra
uw_user	int(11)	NO	PRI	NULL	
uw_wiki	int(11)	NO	PRI	NULL	

Cuadro 4.11: Tabla user_wiki (relaciones de usuarios con wikis)

Field	Type	Null	Key	Default	Extra
wiki_id	int(11)	NO	MUL	NULL	auto_increment
wiki_name	varchar(50)	NO		NULL	
wiki_connection	int(11)	NO		NULL	

Cuadro 4.12: Tabla wiki (wikis del sistema)

4.3. Patrones de diseño (MVC)

Por su amplia aceptación y sus ventajas, se decidió utilizar el patrón MVC o Modelo-Vista-Controlador. Este patrón consiste en dividir la aplicación en tres capas independientes pero conectadas:

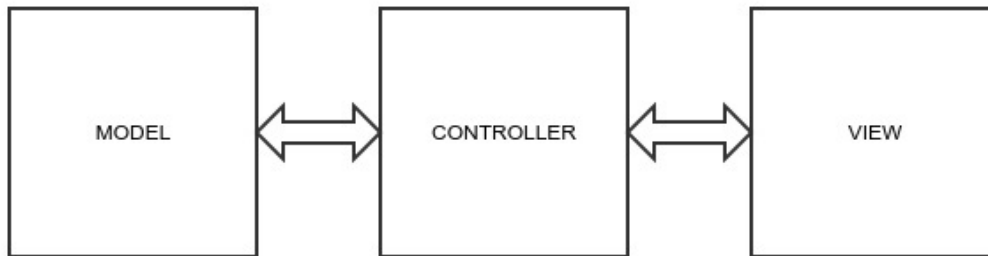


Figura 4.3: Modelo - Vista - Controlador

4.3.1. Modelos

Son los módulos donde se implementarán las funciones de más bajo nivel. Estas serán las encargadas de mantener una conexión bidireccional con las bases de datos y el sistema de ficheros, en caso de ser necesario. Los usuarios nunca interactúan directamente con los modelos. Los modelos implementan la lógica de negocio. En resumen, un modelo definirá una funcionalidad u objeto del sistema y sus interacciones con la infraestructura subyacente.

4.3.2. Vistas

Las vistas son los módulos más cercanos a los usuarios finales. Modelan y crean la interfaz de usuario y son, en definitiva, la parte visible de la aplicación y con la que los usuarios interactuarán. Las vistas no realizan acciones de lógica de negocio por sí misma, sólo definen la parte visual de la aplicación.

4.3.3. Controladores

Los controladores son el nivel intermedio entre las vistas y los modelos. Son los encargados de enviar y recibir datos tanto de los modelos como de las vistas. Son la vía de comunicación entre las “entrañas” del sistema y la parte visual. Cada acción realizada por el usuario será gestionada por los controladores, que llamarán a los modelos para realizar las acciones correspondientes.

4.4. Interfaz

Se diseñó una interfaz simple, fácil de usar y cómoda a la vista. El objetivo principal del diseño fue que el usuario pueda realizar sus tareas sin distracciones, por lo que se eliminó la mayor cantidad de detalles posibles. Todo esto se hizo manteniendo un orden lógico de los elementos en la pantalla, resultando en una interfaz intuitiva, simplista y agradable. Estos fueron los mock-ups inicialmente propuestos:

Configuración	Reportes	Contacto	Sesión
---------------	----------	----------	--------

Opciones	Wikis	Fuentes cualitativas	Usuarios
----------	-------	----------------------	----------

Umbrales

Cálculo de calidad

DEFAULT

Figura 4.4: Mock-up de opciones

Configuración	Reportes	Contacto	Sesión
---------------	----------	----------	--------

Opciones	Wikis	Fuentes cualitativas	Usuarios
----------	-------	----------------------	----------

Wikis

Nombre	Opciones
Wiki 1	[Grupos] [Borrar]
Wiki 2	[Grupos] [Borrar]
Wiki 3	[Grupos] [Borrar]
Wiki 4	[Grupos] [Borrar]

Figura 4.5: Mock-up de wikis

Configuración	Reportes	Contacto	Sesión
---------------	----------	----------	--------

Opciones	Wikis	Fuentes cualitativas	Usuarios
----------	-------	----------------------	----------

Grupos

Crear grupo

Búsqueda de alumnos

Alumno	Grupos	Opciones
Alumno 1	(Grupo 1)	[Asignar grupo]
Alumno 2	(Grupo 4) (Grupo 5) (grupo 6)	[Asignar grupo]
Alumno 3	(Grupo 1) (grupo 3)	[Asignar grupo]
Alumno 4	(Grupo 4) (Grupo 5) (grupo 6)	[Asignar grupo]
Alumno 5	(Grupo 2) (Grupo 3)	[Asignar grupo]

Figura 4.6: Mock-up de grupos

Configuración	Reportes	Contacto	Sesión
---------------	----------	----------	--------

Opciones	Wikis	Fuentes cualitativas	Usuarios
----------	-------	----------------------	----------

Fuentes cualitativas

Nombre	Wiki asociada	Opciones
Fuente 1	Wiki 1	[Borrar]
Fuente 2	Wiki 2	[Borrar]
Fuente 3	Wiki 3	[Borrar]
Fuente 4	Wiki 4	[Borrar]

Figura 4.7: Mock-up de fuentes cualitativas

Configuración	Reportes	Contacto	Sesión
---------------	----------	----------	--------

Opciones	Wikis	Fuentes cualitativas	Usuarios
----------	-------	----------------------	----------

Usuarios

Nombre	Nombre real	Email	Opciones
Usuario 1	Nombre real 1	Email 1	[Borrar]
Usuario 2	Nombre real 2	Email 2	[Borrar]
Usuario 3	Nombre real 3	Email 3	[Borrar]
Usuario 4	Nombre real 4	Email 4	[Borrar]

Figura 4.8: Mock-up de usuarios

Configuración	Reportes	Contacto	Sesión
---------------	----------	----------	--------

Reportes			
Fecha	Wiki	Fuente cualitativa	Opciones
15/07/2020	Wiki 1	Fuente 1	[Borrar] [Ver]
14/07/2020	Wiki 2	NO	[Borrar] [Ver]
13/07/2020	Wiki 1	NO	[Borrar] [Ver]
10/07/2020	Wiki 4	Fuente 4	[Borrar] [Ver]

Figura 4.9: Mock-up de reportes

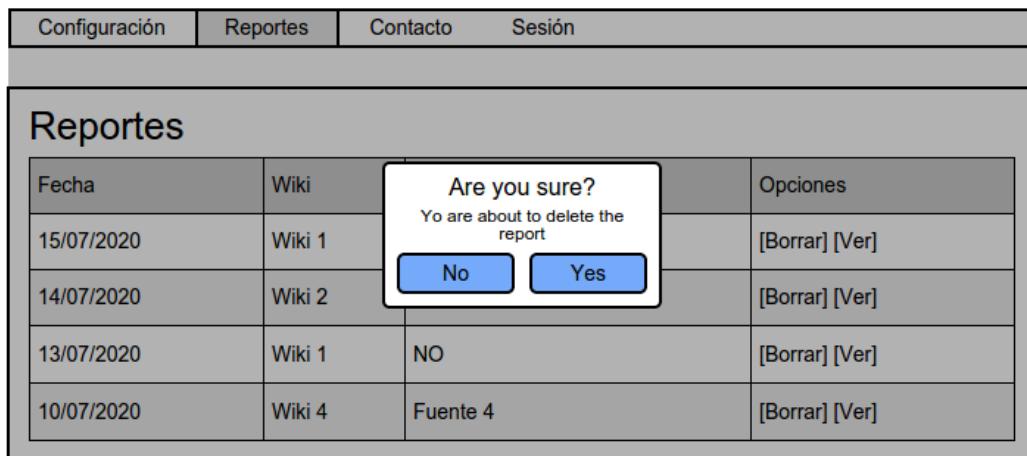


Figura 4.10: Mock-up de popup de borrado

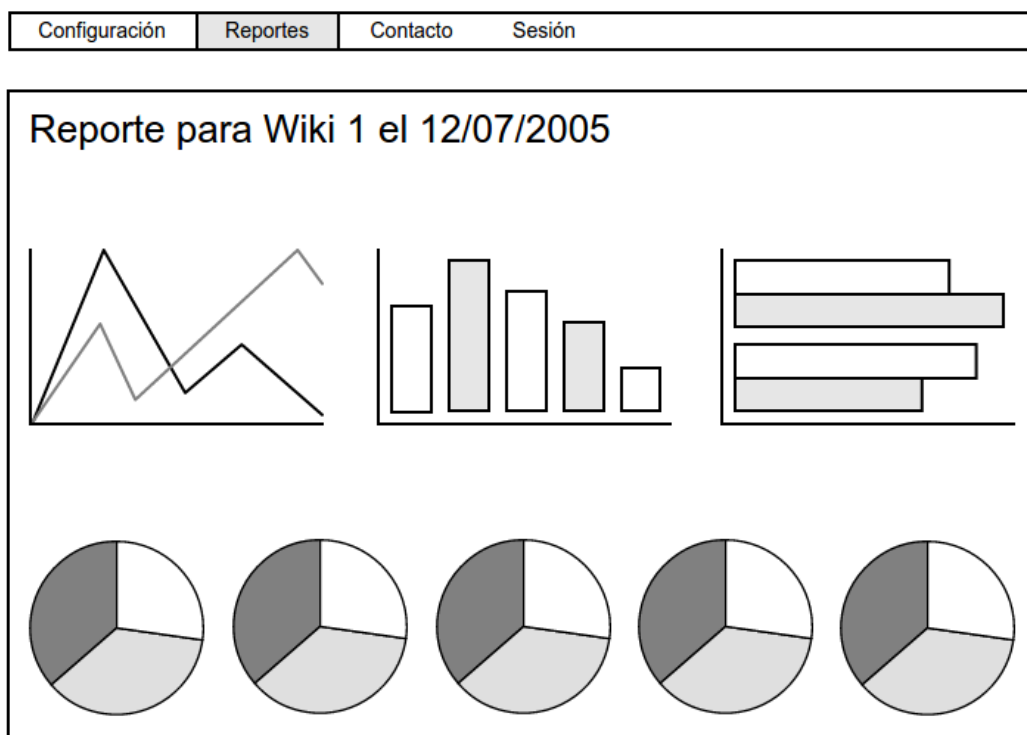


Figura 4.11: Mock-up de vista de reporte

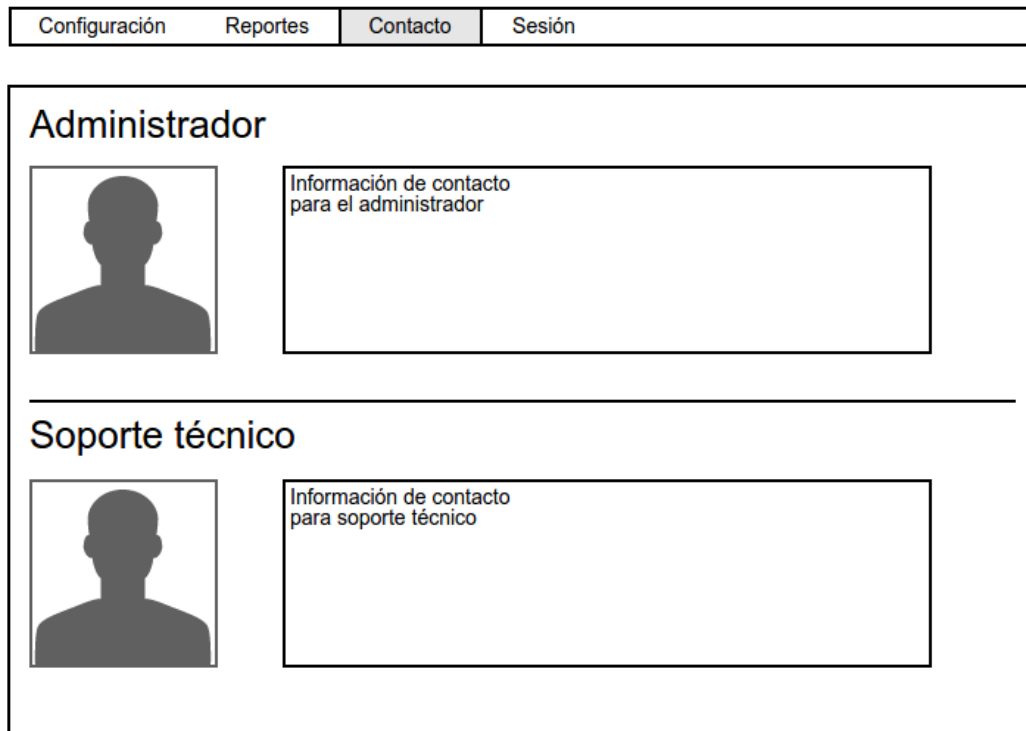


Figura 4.12: Mock-up de contacto

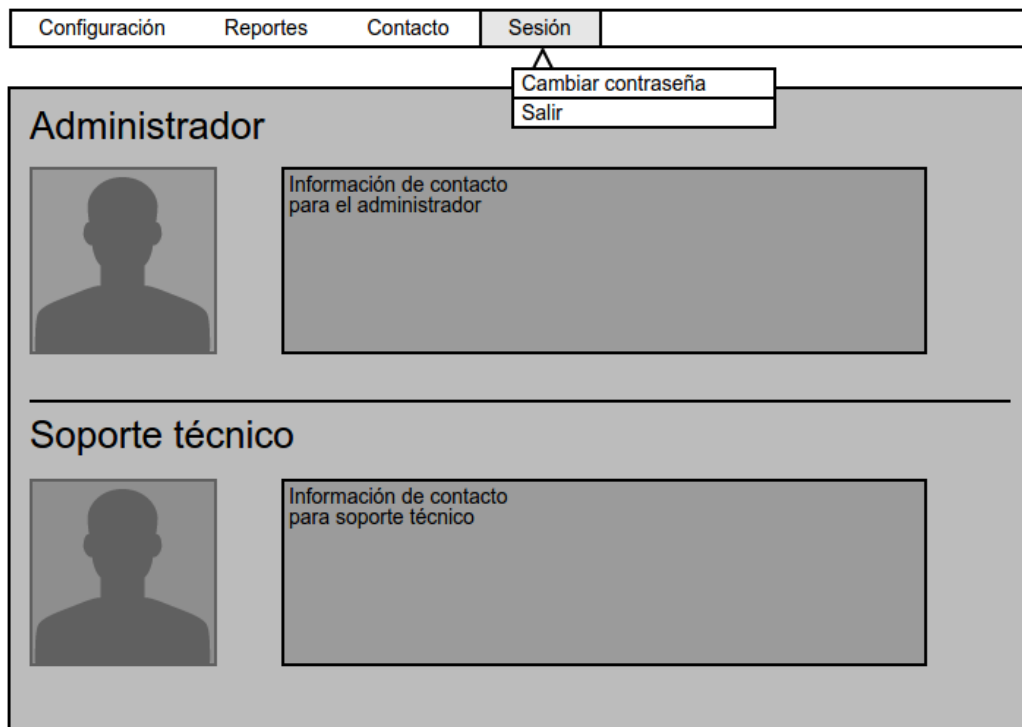


Figura 4.13: Mock-up de menú de header

Capítulo 5

Codificación

En esta sección se dan todos los detalles técnicos relacionados con la implementación del sistema previamente diseñado. Para el desarrollo del proyecto se apostó por herramientas y lenguajes libres, los cuales describimos a continuación. Cabe destacar que, debido a la interrupción del desarrollo por razones personales (reflejado en el diagrama de Gantt adjunto), el proyecto sufrió un rediseño y reimplementación, derivados de mi experiencia profesional adquirida durante esta interrupción.

5.1. Entorno de programación

Para programar se eligió una pila LAMP basada en Debian GNU/Linux. El servidor web sería Apache con su configuración por defecto. Como sistema de gestión de bases de datos se utilizó MySQL y como lenguaje de servidor, PHP [4]. La elección de esta combinación vino dada por tener el alumno experiencia previa con la misma y por su amplio uso en el mundo del desarrollo web. Es un entorno del que mucho se ha investigado y que dispone de una gran cantidad de ayuda y soporte online.[5]

Como editor de código se eligió Vim, un editor basado en terminal, con una larga trayectoria a sus espaldas y con innumerables posibilidades de expansión.

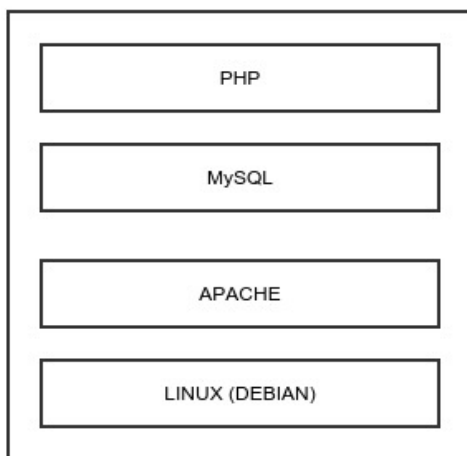


Figura 5.1: Entorno LAMP

5.2. Lenguajes y herramientas

5.2.1. Control de versiones. Subversion.

Para llevar a cabo un desarrollo seguro del proyecto sin preocuparnos por posibles pérdidas del código, se hacía necesario usar un buen sistema de control de versiones. Este nos permitiría llevar un seguimiento del código y poder recuperar versiones antiguas del mismo en caso de que fuera necesario. El sistema elegido fue Subversion. Subversion es un sistema de control de versiones conocido mundialmente y que dispone de un amplio soporte online. Aunque su homólogo más conocido, Git, es hoy por hoy el más usado en el mundo profesional, su curva de aprendizaje es bastante más pronunciada. La principal ventaja de Git radica en su gestión de los conflictos de código. En un proyecto en el que solo iba a codificar una persona, estos conflictos serían inexistentes por lo Git podía suponer una solución “demasiado compleja” y algo complicada. Subversión resultaba más que suficiente para desarrollar el proyecto de forma cómoda y el hecho de que su uso consistiera, en la mayoría de los casos, en tres comandos de terminal consecutivos, facilitó mucho el desarrollo.

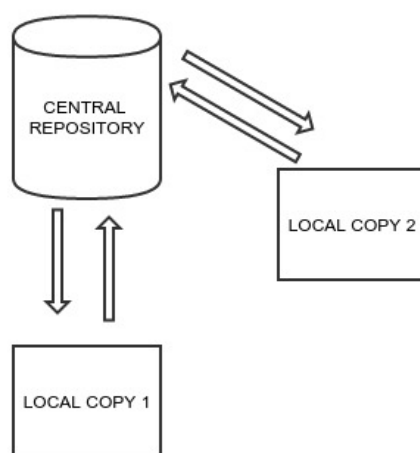


Figura 5.2: Estructura de un repositorio Subversion

Subversion es un sistema de control de versiones centralizado, es decir, solo existe un repositorio central y cada desarrollador tendrá su copia local del código. Cuando un desarrollador suba sus cambios al repositorio central, los demás tendrán que actualizar su copia local para poder subir sus cambios. Si varios desarrolladores trabajan sobre la misma porción de código, uno sube sus cambios y el otro intenta pisarlos, se produce lo que llamamos un conflicto. Como comentamos antes, esto no ocurrirá en este proyecto debido a que solo existe un desarrollador.

Estadísticas del repositorio de CleverFigures

Aquí podemos ver algunas de las estadísticas más significativas del uso del repositorio del proyecto.

Número de commits: 393

Primer commit: 22 de Enero de 2013 (Forja RedIris)

Último commit: 1 de Noviembre de 2014 (SourceForge)

La mayor parte de los commits se dedicaron a la API en primer lugar, y a la interfaz gráfica en segundo, seguidas por las plantillas y las bibliotecas adicionales.

5.2.2. Fuentes de datos. MediaWiki. AssessMediaWiki.

Como ya hemos comentado antes, CleverFigures obtiene sus datos de otros dos sistemas ya implementados: MediaWiki y AssessMediaWiki. Ambos son consultados por CleverFigures directamente en sus bases de datos. Ambas herramientas tienen como sistema de gestión de bases de datos a MySQL por lo que no es necesario separar las aplicaciones en distintos servidores.

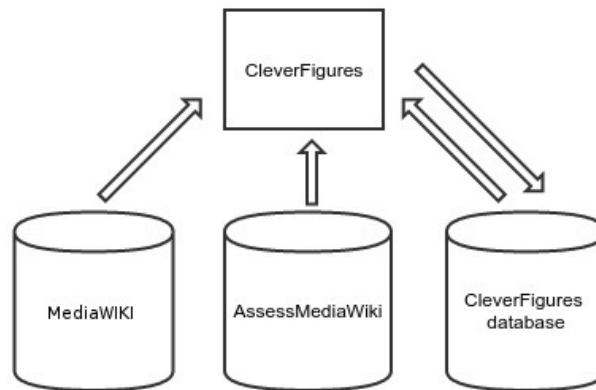


Figura 5.3: Conexiones entre bases de datos de CleverFigures

5.2.3. Base de datos principal. MySQL.

Al disponer de base de datos propia, se hacía necesario escoger un buen sistema de gestión de bases de datos para CleverFigures. MySQL fue la opción escogida. Su elección se basó en el hecho de que es el sistema más usado en proyectos de software libre a nivel mundial y dispone de amplio soporte online. Además, es totalmente gratuita para proyectos sin ánimo de lucro, su instalación es rápida y sencilla y ofrece muchas opciones de configuración y optimización. Sumado a todo esto, también decir que es la herramienta utilizada en la asignatura de bases de datos en la Universidad de Cádiz, por lo que ya resulta familiar y fácil de usar para el alumno desarrollador del proyecto.

Para acceder a las bases de datos usaremos consultas SQL, un lenguaje diseñado para facilitar la obtención de datos desde bases de datos relacionales. Un ejemplo de consulta SQL:

Si dispusiéramos de una base de datos con información sobre el tiempo en distintas ciudades con las fechas de registro

```
SELECT AVG(temperature) FROM weather_registry WHERE month = 7 AND city = 'Granada';
```

nos devolvería la temperatura media histórica de la ciudad de Granada en el mes de Julio.

5.2.4. Lenguaje del servidor (PHP + CodeIgniter)

Como lenguaje de servidor se escogió PHP y su framework CodeIgniter. La elección de PHP viene dada, una vez más, por su uso extendido en el mundo del desarrollo web y por la gran cantidad de frameworks y bibliotecas gratuitas de los que dispone. Uno de estos frameworks, CodeIgniter, fue el elegido para el desarrollo de CleverFigures. Elegimos CodeIgniter por su simplicidad de uso y por facilitar mucho la programación siguiendo el patrón MVC. CodeIgniter dispone de multitud de helpers ya implementados que podemos además combinar con código PHP nativo. Con esto tenemos toda la potencia necesaria para implementar nuestra API de servidor, de forma cómoda y con código de calidad. En nuestro caso, aunque CodeIgniter permite implementar los tres niveles del patrón MVC, decidimos usarlo solo para las capas de Modelo y Controlador. CodeIgniter dispone de un magnífico sistema de templates que nos permitiría implementar también la capa de Vistas, pero decidimos dejar esta capa para el lado del cliente. Los informes tienen una gran carga debido a la visualización de gráficos y dejando la capa de vistas en el lado del cliente conseguimos que el servidor no se sobrecargue generando dichos gráficos y que, además, se generen de forma mucho más rápida, gracias a la posibilidad de generar cada gráfico por separado, de forma asíncrona.

5.2.5. Lenguaje del cliente (Javascript: AngularJS + JQuery)

Para el lado del cliente no tenemos muchas opciones en cuanto al lenguaje a utilizar. JavaScript ha sido y es el lenguaje de lado del cliente con más aceptación y prácticamente el único soportado por los navegadores actuales. Nos permitirá añadir dinamismo a nuestra aplicación web y aumentar el rendimiento de los informes gracias a que nos brinda la posibilidad de llamar de forma paralela y asíncrona al servidor. Usaremos también dos de las bibliotecas más famosas que existen para JavaScript: AngularJS y JQuery. AngularJS es un nuevo framework de lado del cliente desarrollado por Google. Nos permitirá gestionar de forma cómoda las vistas y actualizarlas de forma dinámica con un rendimiento muy alto. AngularJS ofrece, además, una característica llamada data-binding, que nos permite actualizar los datos de las vistas de forma

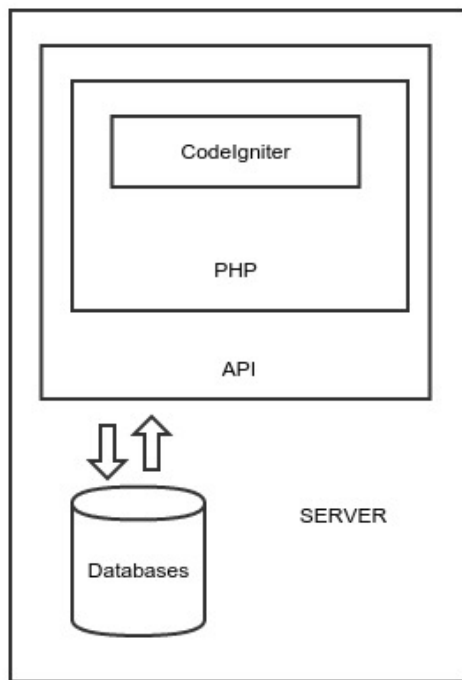


Figura 5.4: Estructura del servidor



Figura 5.5: Logotipo de JavaScript

transparente al programador, sin necesidad de programar funciones que se encarguen de actualizar dichos datos. AngularJS es, sin embargo, una herramienta de reciente creación, y como tal, aún tiene algunas carencias. Para solucionarlas, usaremos JQuery. JQuery es otro framework para JavaScript compatible con AngularJS. Es ya un clásico en la programación web y pondrá a nuestra disposición multitud de funciones que implementan muchas características útiles al programar con JavaScript, como efectos y transiciones.

5.2.6. Graficado. Flot.

Para nuestros informes necesitaremos, por supuesto, generar muchos gráficos. Generar estos gráficos usando código JavaScript nativo podría convertirse en una odisea debido a la gran cantidad de código que supone pintar un simple gráfico. Por ello no dudamos en buscar una buena biblioteca de graficado del lado del cliente. Para no añadir más peso a la aplicación se consideró

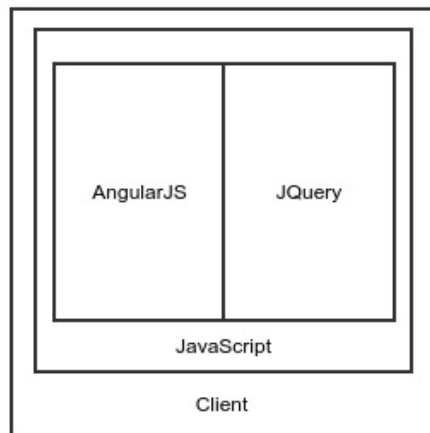


Figura 5.6: Estructura base del cliente

la opción de usar una API online como Google Charts para generar los gráficos, pero pronto nos dimos cuenta de que no era suficiente para lo que necesitábamos. A la hora de decidir qué gráficos queríamos representar, nos dimos cuenta de que Google Charts no era lo suficientemente potente como para generarlos de la mejor forma posible. Es por ello que nos decidimos por Flot. Flot es una biblioteca disponible de forma libre, basada en JQuery. Flot es una de las bibliotecas de graficado más potentes (si no la que más) que existen actualmente para JavaScript y nos ofrecía todo lo que necesitábamos. Tiene infinidad de tipos de gráficos predefinidos y si el que buscamos no lo estaba, se podía diseñar fácilmente. Además de su potencia, su gran comunidad y extensa ayuda online hacen de esta biblioteca una de las más fáciles de usar y una de las soluciones más robustas para la parte más importante de nuestra aplicación.

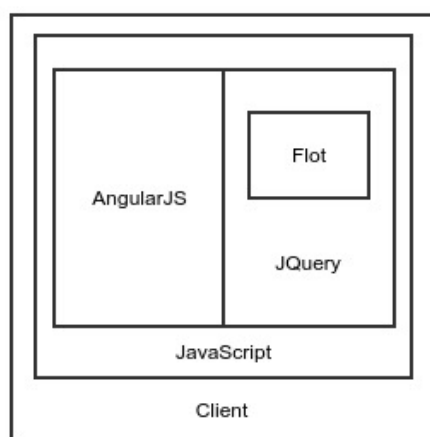


Figura 5.7: Estructura del cliente con Flot

5.2.7. Maquetación. HTML, CSS y Twitter Bootstrap.

Nuestra aplicación no sería nada si, por encima de la lógica de lado del cliente, no dispusiera de una buena maquetación. Las maquetas (vistas, lo que el usuario finalmente ve) de CleverFigures pretenden seguir un diseño simplista y moderno. Se ha utilizado HTML para definir la estructura de las mismas y una pequeña cantidad de código CSS para aplicar algunos estilos personalizados. La mayoría de los estilos de la aplicación son autogenerados con la biblioteca de maquetación Bootstrap. Bootstrap fue creada por Twitter y permite crear interfaces de usuario con un aspecto simplista y actual con simplemente algunas modificaciones en la estructura HTML. El uso de Twitter Bootstrap nos ha ahorrado horas de definición de reglas CSS y funciones JavaScript (es una biblioteca transversal entre la maquetación y la lógica de lado del cliente) y nos ha permitido usar en nuestra aplicación un diseño ampliamente extendido en la web actual y con el que la mayoría de los usuarios estarán familiarizados.

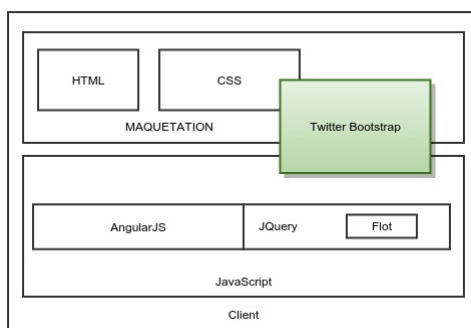


Figura 5.8: Estructura final del cliente

Con esto completamos la descripción de todas las tecnologías que se han utilizado para crear CleverFigures.

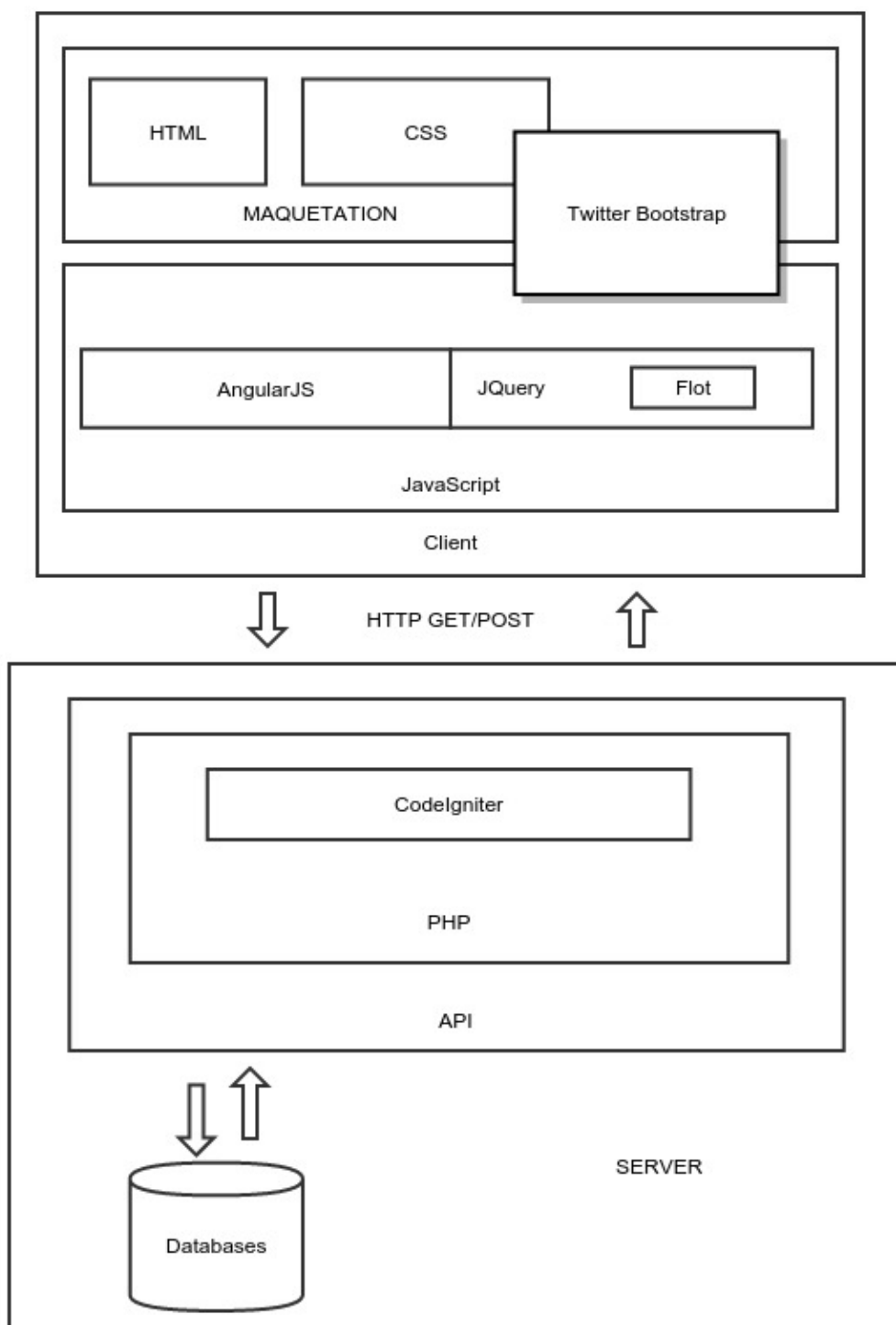


Figura 5.9: Estructura final de CleverFigures

5.2.8. Aspectos relevantes de la implementación

A continuación comentaremos las funcionalidades más destacadas de la aplicación a nivel de implementación. Algunas de estas características fueron requisitos iniciales comentados en las entrevistas y otras surgieron durante la fase de diseño, si bien es verdad, todas influyeron en el hecho de que CleverFigures sea una herramienta de calidad a nivel de código y que el trabajo de los futuros programadores se vea facilitado considerablemente. Otras, como el procesado asíncrono, harán que la experiencia de usuario se incremente de forma notoria.

Obtención y procesado de datos asíncronos

La obtención y procesado de datos para generar los gráficos de los informes se realiza mediante llamadas asíncronas del cliente al servidor. Esto significa que podemos cargar los datos de cada gráfico y generar su visualización de forma independiente y simultánea a los demás gráficos. En las primeras versiones de la aplicación esto se hacía de forma secuencial, lo que afectaba de forma muy negativa al rendimiento. Una revisión de la arquitectura de la aplicación nos hizo darnos cuenta de que podíamos aplicar esta mejora y los resultados fueron inmejorables.

Con la implementación secuencial, cada gráfico esperaba al anterior para generarse. Con la implementación asíncrona, los gráficos se generan simultáneamente y el tiempo de carga se reduce considerablemente.

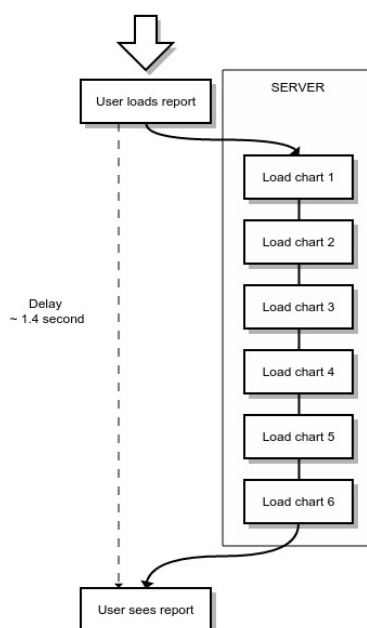


Figura 5.10: Procesado secuencial

Esta técnica no ha sido aplicada exclusivamente a los informes, sino a toda la aplicación. Se ha tenido especial cuidado en la implementación para evitar las posibles condiciones de carrera que pudieran derivarse de su uso. Con ello se ha conseguido disminuir el tiempo de carga medio en un 70 %.

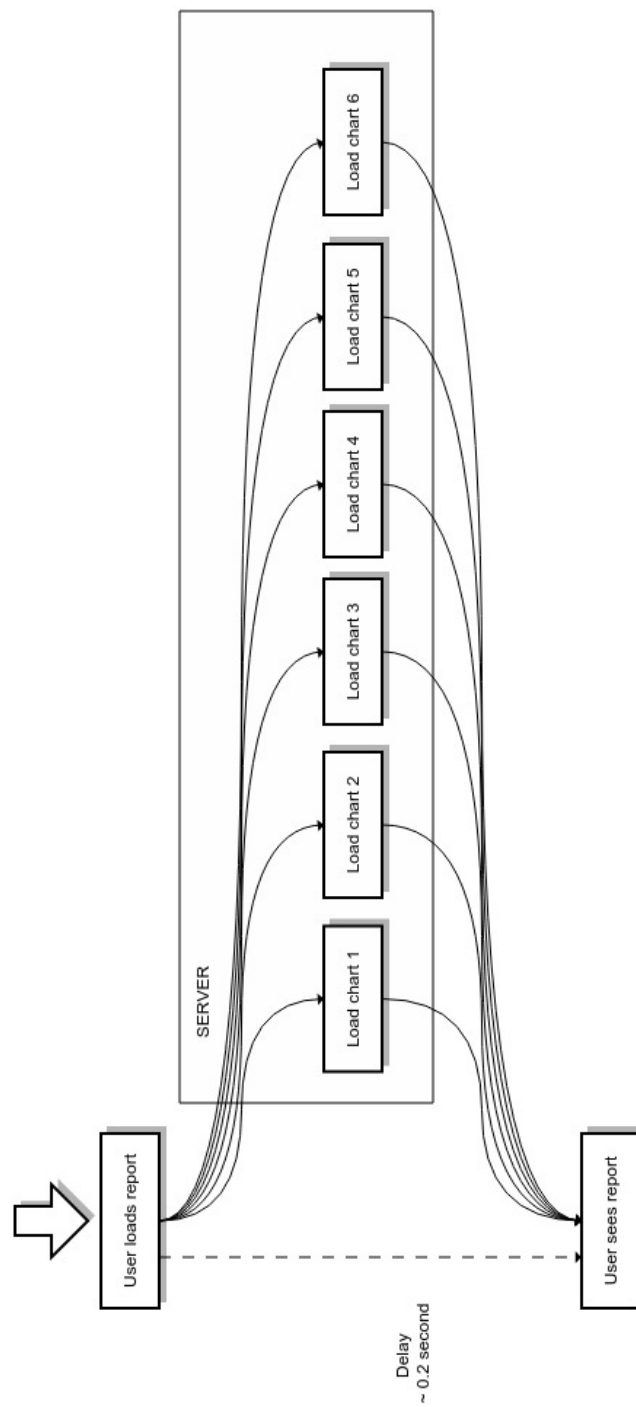


Figura 5.11: Procesado asíncrono

API

Una de las principales características de CleverFigures de cara a futuros desarrolladores es que todas las funcionalidades en la parte del servidor están accesibles a través de una API JSON, implementada a nivel de controlador. El propio front-end de la aplicación consume de dicha API para implementar la interfaz, por lo que la aplicación queda lista para el usuario final. Sin embargo, si alguien desea implementar su propia interfaz o recibir los datos de los informes directamente en una terminal, puede hacerlo sin problemas a través de la API. Entre las funciones más destacadas de las que disponemos en la API se encuentran las tareas de administración, la gestión de análisis y de fuentes de datos, y la recepción de datos en formato JSON, correspondiente a uno o varios gráficos.

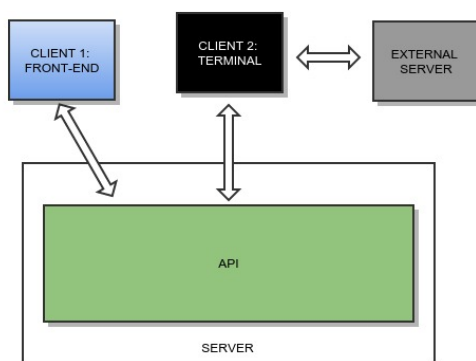


Figura 5.12: Estructura de la API de CleverFigures

Acceso total a base de datos de MediaWiki

MediaWiki pone una API programática a disposición de los desarrolladores para facilitar la conexión con las wikis y posibilitar la lectura y escritura de datos en la wiki. En el caso de CleverFigures, hemos optado por no usar esta API y conectarnos directamente a la base de datos de las wikis. Una API supone una capa sobre la base de datos real, que frecuentemente imposibilita realizar ciertas acciones sobre la misma. CleverFigures está diseñado para ser usado con wikis sobre las que tengamos control, y por lo tanto, acceso total. Para realizar la conexión con la wiki, simplemente hay que proporcionar a CleverFigures un usuario en la base de datos, así como la localización de la misma. Con esto conseguimos total flexibilidad en las operaciones a realizar con la wiki, y abrimos todas las puertas a futuras implementaciones, tanto de lectura y recopilación de datos, como de escritura sobre los propios artículos.[3]

Cálculo personalizado de la calidad

Durante el desarrollo de la aplicación, y más concretamente del módulo encargado de gestionar las estadísticas cualitativas, nos dimos cuenta de que “calidad” puede no siempre significar lo mismo. Dependiendo del tipo de trabajo realizado por el alumno, o de los criterios utilizados, un “trabajo de calidad” puede ser definido de formas distintas, y por lo tanto, no hay una única forma de evaluar la calidad. Hemos querido tener esto en cuenta en CleverFigures. En la sección de configuración, CleverFigures permite elegir cómo se calculará la calidad. Lo único que tiene que hacer el profesor es definir, de forma matemática lo que quiere hacer con sus datos. Dos variables

estarán disponibles para este cálculo: la cantidad de bytes aportados (es decir, la cantidad de texto) y la nota recibida. De esta forma, cada profesor puede definir qué considera un trabajo de calidad. Unos ejemplos:

$$Y = Bytes + Nota * 1000$$

Con esta función, podremos ver una gráfica que sube considerablemente cuando la nota recibida es alta.

$$Y = Bytes + (Nota - 5) * Bytes$$

Con esta función, obtendremos una gráfica que sube (más o menos, dependiendo de la nota) cuando la nota es mayor que cinco y que baja cuando es menor.

Esto son solo dos ejemplos, pero observamos que esta funcionalidad nos da total flexibilidad y permite a los docentes reflejar la realidad de forma más fiel, al poder definir exactamente cómo quieren evaluar.

Integración con distintas fuentes de datos cualitativas

Aunque CleverFigures usa actualmente AssessMediaWiki como fuente de datos cualitativa, entendemos que es posible que existan bases de datos con información cualitativa recopilada durante años y que no usen este software. Se hace necesario entonces disponer de una forma de extraer estos datos para usarlos en los informes, en caso de que sea necesario. Para facilitar esto a los futuros desarrolladores, CleverFigures define una interfaz, esto es, una colección de funciones, que, una vez definidas, permiten obtener los datos de una fuente distinta a AssessMediaWiki. En el caso de que un nuevo desarrollador quiera conectar CleverFigures con una nueva fuente de datos, solo tendrá que implementar las funciones que esta interfaz declara. Una vez que tenga estas funciones definidas, CleverFigures entenderá cómo debe conectarse a esta nueva fuente de datos y podrá realizar informes usando los datos existentes en ella. Siguiendo este procedimiento, podemos conectar CleverFigures a cualquier base de datos que necesitemos.

Capítulo 6

Pruebas

Como en cualquier otro proceso de ingeniería, en el desarrollo de software necesitamos disponer de medidas y procedimientos que nos aseguren que el producto final que hemos creado es de calidad, hace lo que necesitamos y no va a causarnos problemas no contemplados. En el desarrollo de `CleverFigures` hemos seguido diversas medidas de aseguramiento de la calidad (o más comúnmente *Quality Assurance*), las cuales explicamos a continuación.

6.1. Pruebas unitarias en la API

Para asegurar la calidad del código del servidor, se ha procedido a probar cada función definida de forma unitaria. Las pruebas o tests unitarios consisten en asegurar que una función de código hace lo que debe en todas las situaciones posibles en las que puede ejecutarse. Idealmente, deberíamos probar cada función con todos los tipos de entrada posibles para asegurar que la misma siempre se va a comportar de forma correcta. En la práctica, esto no es posible en la mayoría de los casos y se suelen reducir las pruebas a un conjunto de casos razonable. Para el código de servidor de `CleverFigures`, se ha elaborado un plan de pruebas que garantiza el buen funcionamiento de las funciones definidas, siempre a nivel unitario. Para elaborar el plan de pruebas unitarias, se siguieron las siguientes reglas:

- Si la función probada tiene parámetros por defecto, es necesario probarla con al menos una combinación de parámetros de entrada vacíos (aquí representados con `null`).
- Se debe probar al menos una combinación de parámetros inválida por cada caso negativo de la función.
- Se debe probar al menos una combinación de parámetros válida por cada caso positivo de la función.

El plan de pruebas elaborado sirvió para detectar diversos errores, la mayoría de ellos en las funciones de graficado de la API. Su ejecución se repitió varias veces durante el periodo de pruebas conforme se arreglaban los errores, para garantizar el buen funcionamiento de las funciones en cuestión a nivel unitario.

Podemos ver el banco de pruebas unitarias en el apéndice de la página 105.

6.2. Pruebas funcionales

El hecho de tener un código de calidad a nivel unitario no siempre garantiza que la aplicación final haga lo que tiene que hacer. Es necesario ver cómo todas las funciones implementadas trabajan juntas y consiguen, finalmente, llevar a cabo las tareas o funcionalidades que el cliente, o en nuestro caso el personal docente, necesita. Para esto hemos elaborado un plan de pruebas funcionales que garantizan que la aplicación trabaja como debe a nivel funcional, y que, por lo tanto, cumple los requisitos funcionales acordados inicialmente. En dicho plan de pruebas se cubren de forma exhaustiva todos los casos de uso de los que hablamos en la sección de especificación.

Podemos ver el banco de pruebas funcionales en el apéndice de la página 132.

Parte III

Epílogo

Capítulo 7

Manual de instalación

La instalación de CleverFigures es sencilla y rápida. A continuación detallamos cómo instalar todos sus componentes:

7.1. Pre-requisitos

- Es necesario tener un servidor web instalado en el servidor en el que se vaya a instalar CleverFigures. En concreto, CleverFigures ha sido desarrollado y probado sobre **Apache**.
- Es necesario disponer de una versión de **PHP** ≥ 5.4 instalada en el servidor.
- Para descargar el repositorio de CleverFigures, es necesario tener instalado **Subversion** en el servidor.
- CleverFigures funciona con **MySQL**. Es necesario tenerlo instalado para poder hacer la instalación de la base de datos.

7.2. Pasos de la instalación:

1. Descargar el repositorio de CleverFigures con el siguiente comando en el directorio de nuestro servidor web:

```
svn checkout https://svn.code.sf.net/p/cleverfigures/code/ cleverfigures-code
```

2. Para definir la información de contacto es necesario editar el fichero

```
<carpeta\_instalación>/trunk/frontend/app/config/localsettings.js
```

y rellenar las variables con prefijo **admin_** y **technical_support_**.

3. Debemos asegurarnos de que tenemos activado **mod_rewrite** en nuestra instalación de PHP.
4. Iniciar sesión en MySQL y crear la base de datos con la que trabajará CleverFigures. Puede encontrar el fichero de creación de la base de datos en el directorio

```
<carpeta\_instalación>/trunk/backend/sql.
```

5. Editar el fichero

`<carpeta_instalación>/trunk/backend/config/database.php`

y rellenar la información de las variables `hostname`, `username`, `password` y `database` del array con la configuración de la base de datos.

6. Inserta en MySQL el usuario Admin para poder acceder a la aplicación por primera vez:

```
INSERT INTO user
SET
  user_username = 'admin',
  user_password = MD5('<tu\_contraseña\_aquí>'),
  user_realname = '<tu\_nombre\_aquí>',
  user_email = '<tu\_mail\_aquí>',
  user_language = 'spanish',
  user_is\_admin = 1
;
```

Una vez seguidos estos pasos, CleverFigures debería estar listo para su uso. Accede desde tu navegador a

`<carpeta_instalación>/trunk/frontend/app/`

y deberías ver la pantalla de login. Ahí puedes escribir las credenciales que hayas puesto en el usuario en el último paso de la instalación.

Capítulo 8

Manual de usuario

8.1. Sesión

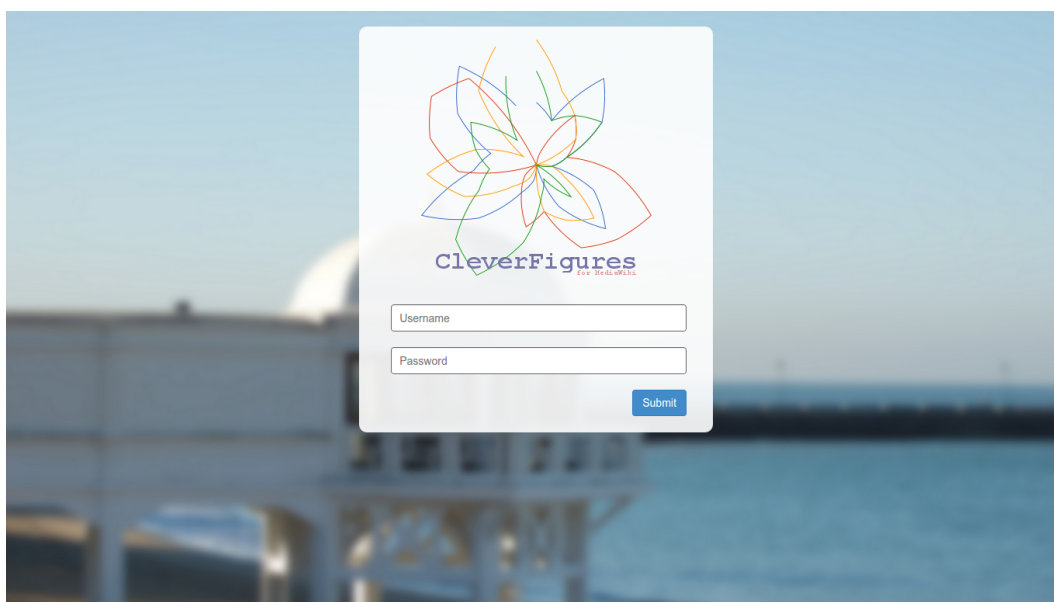


Figura 8.1: Pantalla de inicio de sesión

Para poder usar CleverFigures es necesario estar registrado en el sistema y por lo tanto disponer de un usuario y contraseña válidos. Una vez se disponga de ellos (para lo cual será necesario contactar con un administrador) sólo será necesario acceder a la pantalla de login e introducirlos en los respectivos campos “Nombre de usuario” y “Contraseña”. Pulsando el botón Enviar, la aplicación abrirá una sesión para el usuario y le permitirá acceder a la lista de reportes. Una vez abierta la sesión, para cerrarla solo hay que pulsar Salir en el menú desplegable de la barra superior.

8.2. Cambio de contraseña

Una vez haya iniciado la sesión, es posible para el usuario cambiar la contraseña que usa para acceder. Para hacer esto se le pedirá introducir la contraseña actual, la nueva, y confirmar la

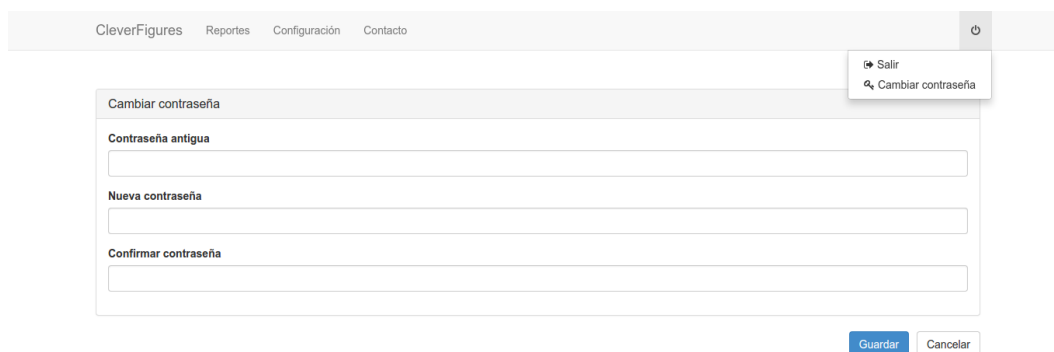


Figura 8.2: Pantalla de cambio de contraseña

nueva escribiéndola de nuevo. El formulario de cambio de contraseña está disponible a través de la opción Cambiar contraseña del menú desplegable de la barra superior.

8.3. Reportes

Los reportes son la parte más importante de CleverFigures. Mediante ellos podemos ver el estado de nuestras wikis y evaluar el trabajo de los alumnos. A continuación se explican las principales funciones que podemos hacer con respecto a los reportes.

8.3.1. Listado de reportes

Si pulsamos Reportes en el menú superior, accederemos al listado de reportes. En dicha vista podemos ver todos los reportes que hayamos generado previamente. De cada uno de ellos, podemos ver la wiki sobre la que se realizó, si se usó una fuente cualitativa y cuál, la fecha y hora del reporte y las acciones que tenemos disponibles para cada uno de ellos.

8.3.2. Creación de reportes

En la vista de listado de reportes, disponemos de un botón para generar reportes. Está localizado arriba a la derecha y nos permitirá crear nuevos reportes sobre los datos que nos interesen. Una vez pulsamos el botón, aparece un popup en el que podremos seleccionar la wiki del reporte y, opcionalmente, una fuente cualitativa. Después pulsamos Crear reporte y automáticamente después ya tendremos nuestro reporte listo para ser visualizado.

8.3.3. Visualización de reportes

Una vez generado el reporte, para visualizarlo solo tenemos que volver al listado de reportes. Ya allí, elegimos el reporte que queremos visualizar, y hacemos click en el icono del ojo, en la

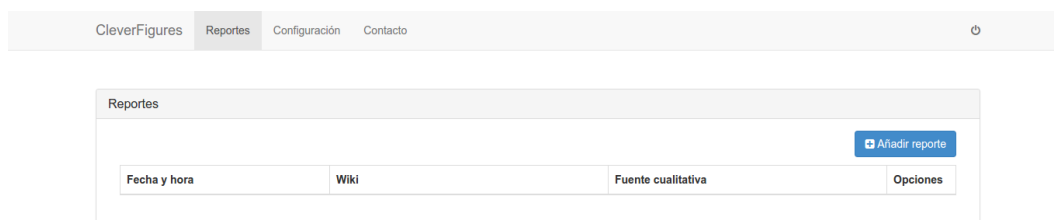


Figura 8.3: Pantalla de listado de reportes

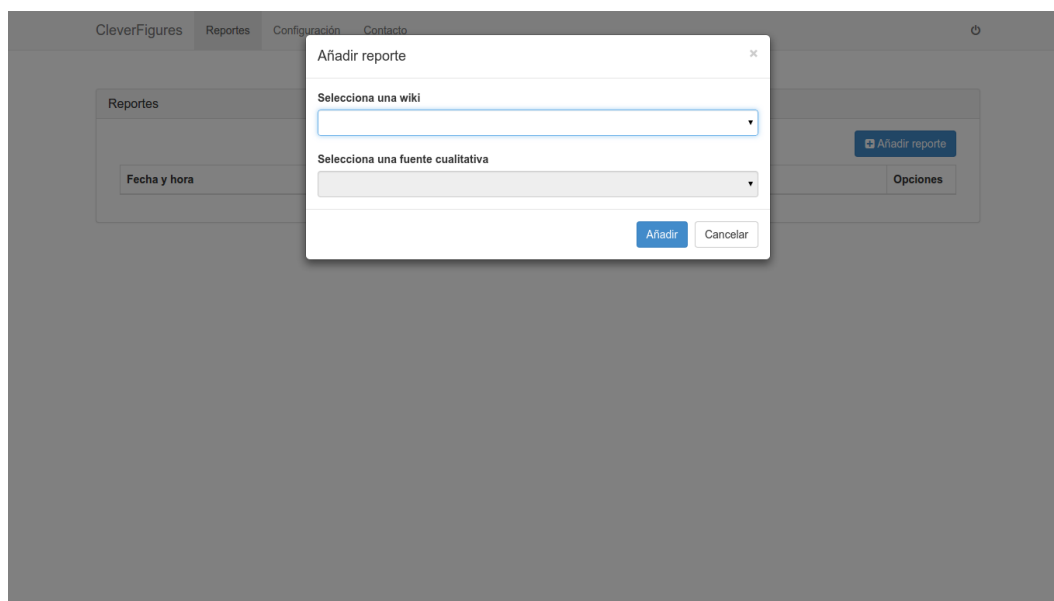


Figura 8.4: Pantalla de creación de reportes

columna Opciones. El sistema nos llevará al reporte general de la wiki, en el que podremos ver los distintos gráficos que reflejan el estado de la wiki para la fecha escogida. Desde el reporte general podremos navegar a los reportes de páginas, usuarios, categorías y grupos, haciendo click en los nombres que aparecen en las distintas tablas del reporte.

8.3.4. Borrado de reportes

Si queremos borrar un reporte, podemos hacerlo también desde el listado de reportes. Esta vez, usaremos el icono de la papelerita disponible en la columna de Opciones. Al hacer click en el

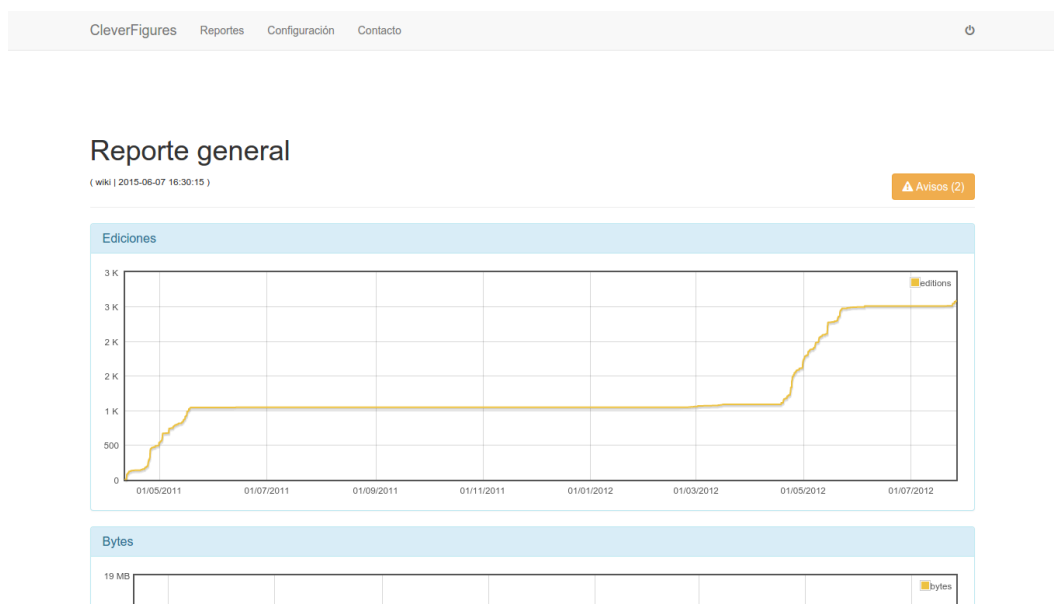


Figura 8.5: Pantalla de visualización de reportes

mismo se nos pedirá confirmación del borrado. Desde el popup podemos cancelar el borrado si cambiamos de opinión, o confirmarlo, si finalmente decidimos eliminarlo.

8.4. Configuración

The screenshot shows the 'Configuración' page. The navigation bar at the top includes 'CleverFigures', 'Reportes', 'Configuración', and 'Contacto'. Below the navigation bar, there are tabs for 'Opciones', 'Wikis', 'Fuentes cualitativas', and 'Usuarios'. The 'Opciones' tab is selected. The main content area is divided into two sections. The first section, 'Umbral de aviso', contains six sliders for setting thresholds: 'Rango de ediciones para wikis' (0 to 500), 'Rango de ediciones para páginas' (0 to 500), 'Rango de ediciones para usuarios' (0 to 500), 'Rango de bytes para wikis' (0 to KB), 'Rango de bytes para páginas' (0 to 5 KB), and 'Rango de bytes para usuarios' (0 to 10 KB). The second section, 'Opciones de fuentes cualitativas', contains a text input field for 'Función de calidad'. A 'Guardar' button is located at the bottom right of the page.

Figura 8.6: Pantalla de configuración

En este apartado veremos las diferentes formas de configurar CleverFigures para adaptarlo de la mejor forma al uso que vamos a darle. Existen cuatro apartados de configuración: Opciones,

Wikis, Fuentes cualitativas y Usuarios. La última solo está disponible para los administradores, mientras que el resto es accesible a cualquier usuario de la plataforma. A todas estas configuraciones podemos acceder haciendo click en el enlace Configuración del menú superior.

8.4.1. Opciones

En esta pestaña tenemos disponibles la configuración de umbrales de aviso y de la función de calidad. Los umbrales de aviso son límites que podemos configurar y que determinarán cuándo queremos ser alertados. Los usuarios añaden contenido a la wiki en mayor o menor medida y quizá queremos ver un aviso cuando un usuario no añade suficiente contenido, o cuando, por el contrario, ha añadido mucho. Hay varios tipos de aviso como son los bytes añadidos, la actividad del usuarios, etc. Por otro lado, podemos establecer también qué función de calidad queremos usar. Esta función determina cómo se calculan los gráficos mixtos cuantitativos-cualitativos, de forma que podamos adaptar el resultado a nuestra forma de evaluar. Si no añadimos ninguna, se usará la función por defecto del sistema.

8.4.2. Wikis

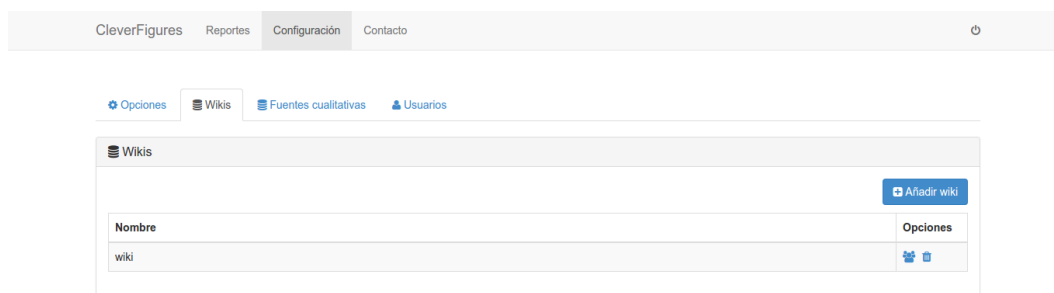


Figura 8.7: Pantalla de configuración de wikis

En esta pestaña podemos realizar todas las operaciones de administración relacionadas con las wikis. Podemos ver las wikis que tenemos en el sistema, añadir más, borrar las que no necesitamos o gestionar sus grupos de alumnos.

8.4.3. Visualización de wikis

Para ver el listado de wikis disponibles solo hay que navegar a la tabla de wikis de la misma pestaña.

8.4.4. Borrado de wikis

Para borrar una wiki, debemos buscarla en la tabla de wikis disponibles y hacer click en el icono de la papelera que encontraremos en la columna de Opciones. Se nos pedirá confirmar el borrado, para evitar borrados accidentales.

8.4.5. Gestión de grupos

Los alumnos registrados en una wiki pueden ser agrupados para poder hacer evaluaciones grupales. Para gestionar estos grupos, debemos ir a la tabla de wikis, buscar la wiki cuyos alumnos queremos agrupar, y hacer click en el icono de Grupos de la columna Opciones. Una vez dentro de la pantalla de gestión de grupos, podemos realizar varias acciones: añadir grupos, para lo cual deberemos especificar un nombre para el grupo; asignar grupos a alumnos, o agrupar los alumnos de forma automática según el trabajo que han hecho. Esta última funcionalidad resulta muy útil cuando los alumnos han empezado a trabajar y queremos registrar todos los grupos. El sistema lo hará automáticamente por nosotros.

8.4.6. Fuentes cualitativas

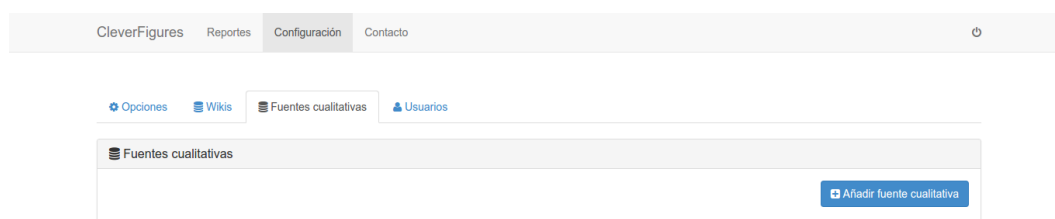


Figura 8.8: Pantalla de configuración de fuentes cualitativas

En esta pestaña podemos realizar todas las operaciones de administración relacionadas con las fuentes cualitativas. Podemos ver las fuentes que tenemos en el sistema, añadir más o borrar las que no necesitamos.

8.4.7. Visualización de fuentes cualitativas

Para ver el listado de fuentes disponibles solo hay que navegar a la tabla de fuentes cualitativas de la misma pestaña.

8.4.8. Borrado de fuentes cualitativas

Para borrar una fuente, debemos buscarla en la tabla de fuentes disponibles y hacer click en el icono de la papelera que encontraremos en la columna de Opciones. Se nos pedirá confirmar el borrado, para evitar borrados accidentales.

8.4.9. Usuarios

Esta pestaña solo se encuentra disponible para los administradores. Desde ella podremos gestionar los usuarios del sistema.

8.4.10. Visualización de usuarios

Para ver el listado de usuarios solo hay que navegar a la tabla de usuarios de la misma pestaña.

8.4.11. Adición de usuarios

Podemos añadir usuarios al sistema completando el formulario de adición de usuarios que encontraremos al pulsar el botón Añadir usuario en la pestaña de gestión de usuarios. Una vez estén rellenos los datos, pulsamos Guardar y el usuario quedará registrado. Después de registrar un nuevo usuario, podemos darle sus credenciales de acceso y recomendarle, por seguridad, que cambie su contraseña una vez acceda al sistema.

8.4.12. Borrado de usuarios

Para eliminar un usuario del sistema, debemos navegar hasta la lista de usuarios disponibles y hacer click en el icono de la papelera en la columna de opciones. El usuario dejará de tener acceso al sistema y todos los datos relacionados con él serán borrados.

8.4.13. Funciones de calidad

Como administrador, podemos añadir funciones de calidad al sistema, las cuales estarán a disposición de todos los usuarios para usarlas en sus reportes. Para añadir una función es necesario editar el archivo de configuración de la API, y añadir a la lista de funciones de calidad la nueva función. Su definición debe ser escrita en lenguaje PHP, por lo que se recomienda tener un mínimo de experiencia con dicho lenguaje si se va a editar este fichero.

8.4.14. Datos de contacto

Los datos de contactos se pueden añadir o modificar en el archivo de configuración de la aplicación cliente. Una vez que los añadamos allí, sólo debemos guardar el fichero y refrescar el navegador para ver los cambios.

Capítulo 9

Conclusiones

9.1. Conclusiones personales sobre el proyecto

Este proyecto me ha enriquecido a varios niveles. Su duración me ha servido para ver que en el campo de la Informática, todo evoluciona, evoluciona rápido y evoluciona siempre. No importa con qué tecnologías empecemos un proyecto. Si este se extiende más de varios meses, nuevas versiones habrán aparecido y nos daremos cuenta de que nuestro proyecto está, desde su nacimiento, obsoleto. Esto no significa que no sea útil, simplemente no usa las tecnologías más actualizadas. Por otro lado, el hecho de trabajar en el mundo del desarrollo de software al mismo tiempo que desarrollaba el proyecto, me ha hecho más consciente de mi evolución personal. El proyecto sufrió un rediseño y reimplementación a mitad de su desarrollo por este mismo motivo. Cuando miras al código que desarrollaste hace un año, debe (y esto es pura opinión personal) invadirte una sensación mezcla de vergüenza y compasión por los errores que cometías y los procedimientos inadecuados que seguías. Esto es el mejor signo de que ahora eres mejor, profesionalmente hablando. En mi caso, todo esto se ha reflejado en una aplicación que, creo, tiene un código de mucha calidad, de cuya estructura estoy orgulloso y ya espero con ansias ese momento en el que pueda volver a sentir que he evolucionado. En cuanto a la aplicación, estoy muy contento por lo útil que puede llegar a ser y por el trabajo que puede llegar a ahorrar. Esta fue mi primera aplicación, y me siento muy orgulloso de que haya sido en el ámbito del software libre y sobre todo, de que esté dedicada al mundo de la educación. Si mi trabajo sirve para que los procedimientos mejoren a la hora de evaluar, aunque sea solo un poco, no puede ser malo.

9.1.1. Relación calidad - hora en el trabajo del alumnado

En la búsqueda de nuevas formas de representar la cantidad masiva de datos que teníamos a mano, probamos muchísimas combinaciones para lograr gráficos informativos que ofrecieran una nueva visión del trabajo que los alumnos estaban realizando. La gráfica de esta sección, por ejemplo, relaciona la hora de la edición (es decir, la hora a la que el alumno realizó su aportación a la wiki) con la nota media recibida por parte de sus compañeros en dichas ediciones. Aunque el gráfico carece de línea de tendencia, si observamos los puntos la progresión es clara: las ediciones entre las 18h y las 00h reciben, de media, un punto más que las realizadas entre las 06h y las 12h. En términos más simples, los alumnos analizados, trabajaban mejor si lo hacían por la tarde-noche. Igualmente se podrían buscar otros perfiles, como los detectados en [8].

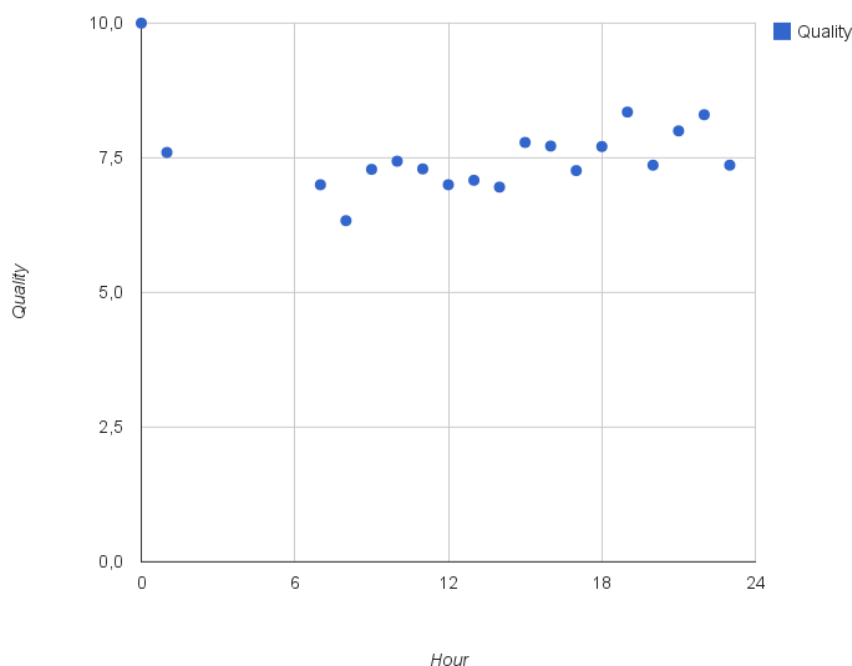


Figura 9.1: Relación calidad - hora

9.1.2. Concurso de software libre

El Concurso Universitario de Software Libre (CUSL) es un concurso a nivel nacional, que se celebra anualmente. En él se reconoce a los mejores proyectos universitarios elaborados exclusivamente usando software libre. CleverFigures fue presentado a la edición de 2012 de dicho certamen, con resultados más que satisfactorios. Ganó el primer premio en la fase local de la Universidad de Cádiz. La ceremonia de entrega de premios tuvo lugar el 20 de Marzo de 2013 en la facultad de Filosofía y Letras, y quedó reflejada en esta noticia de la Oficina de Software Libre de la Universidad de Cádiz [10]. En ámbito nacional, el proyecto obtuvo una mención honorífica y fue presentado mediante un vídeo proyectado en la Fase Final, que se celebró en la Universidad de Granada [6][7].

9.1.3. Ampliaciones y mejoras

Aunque podemos decir que el proyecto está terminado y se han cumplido los requisitos iniciales, un proyecto de software, en la mayoría de los casos, siempre será ampliable con mejoras. CleverFigures no es una excepción y hemos elaborado una lista de futuros añadidos que darían un plus de calidad al sistema.

Integración con AssessMediaWiki

CleverFigures tiene, actualmente, una fuerte dependencia de AssessMediaWiki, al ser su principal fuente de datos cualitativos. Las funciones que realizan AssessMediaWiki y CleverFigures no

son mutuamente excluyentes y se podrían integrar ambos en la misma aplicación, de forma que los alumnos pudieran acceder a CleverFigures para realizar la evaluación por pares. Tanto CleverFigures como AssessMediaWiki están desarrollados sobre CodeIgniter, lo que facilitaría su integración.

Implementación como extensión de MediaWiki

MediaWiki dispone de un repositorio de extensiones muy amplio. Convertir CleverFigures en una extensión facilitaría su instalación sobre cualquier wiki. Para ello sería necesario adaptar el modelo de datos para trabajar con una sola wiki en vez de múltiples. Así mismo, las extensiones de MediaWiki tienen una estructura de ficheros determinada que habría que respetar. Este trabajo no sería fácil ya que requeriría de múltiples modificaciones en el código de CleverFigures, y un posterior testeo en profundidad, pero harían que la aplicación ganara en facilidad de instalación y visibilidad.

Referencias

[1] *El futuro de método docente. Un nuevo enfoque.*

Juan Ortega-Valiente y Antonio J. Reinoso. 2013. Revista Tecnol@ y desarrollo. Escuela Politécnica Superior. Universidad Alfonso X el Sabio. 28691, Villanueva de la Cañada (Madrid). ISSN: 1696-8085.

<http://www.uax.es/publicacion/el-futuro-del-metodo-docente-universitario-un-nuevo-enfoque-ba.pdf>

[2] *Uso de wikis para la realización de trabajos colaborativos en el aula.*

González Pareja, A., Calderón Montero, S., Galache Laza, T. y Torrico González, A. 2006. XIV Jornadas de ASEPUMA y II Encuentro Internacional.

<http://www.uv.es/asepuma/XIV/comunica/118.pdf>

[3] *Wikimedia architecture.*

Mark Bergsma. Wikimedia Foundation Inc. 2013.

<https://std-svn-uppa.googlecode.com/svn/trunk/SIGLIS%202012%20-%202013/M2/Urbanisation/cours/architecture%20technique/wikimedia-architecture.pdf>

[4] *PHP.net*

The PHP Group. 2001-2015.

<https://php.net/>

[5] *StackOverflow.com*

Stackexchange Team. 2015.

<https://www.stackoverflow.com>

[6] *Noticia fase final VII CUSL*

Oficina de Software Libre de la Universidad de Zaragoza. 2013.

<http://osluz.unizar.es/content/premiados-de-la-fase-final-del-vii-cusl>

[7] *Dos estudiantes de la UCA, entre los premiados en el 7º Concurso Universitario de Software Libre*

Universidad de Cádiz. 2013.

<http://www.uca.es/es/cargarAplicacionNoticia.do?identificador=5944>

[8] *Similarities, challenges and opportunities of Wikipedia content and open source projects.*

Andrea Capiluppi. 2012. Journal of Software: Evolution and Process. Volume 25. Páginas 891-914.

<http://onlinelibrary.wiley.com/doi/10.1002/smr.1570/abstract>

[9] *Gestión documental con plataformas wiki.*

Jesús Tramullas. 2008. Congreso Internacional de Información INFO 2008.

[10] *Ganadores de la fase local del CUSL7*

Oficina de Software Libre de la Universidad de Cádiz. 2013.

<http://osl.uca.es/node/1232>

[11] *Esquema de base de datos de AssessMediaWiki*

Álvaro Almagro Doello, Antonio Balderas Alberico, Manuel Palomo Duarte, Alberto García y Juan Manuel Dodero Beardo. 2012.

<https://forja.rediris.es/projects/assessmediawiki/>

[12] *Diagrama de clases oficial de MediaWiki*

Wikimedia Foundation. 2014.

https://upload.wikimedia.org/wikipedia/commons/f/f7/MediaWiki_1.24.1_database_schema.svg

[13] *Fundamentos de Algoritmia*

Giles Brassard y Paul Bratley. 1997. Prentice Hall.

[14] *Algorithm Design*

Jon Kleinberg y Eva Tardos. 2005. Pearson. First edition.

[15] *Estructuras de datos. Teoría.*

Martínez, P. Gallardo C. y Sánchez, J. 2010. Servicio de Publicaciones EUI. Universidad Politécnica de Madrid.

[16] *Prácticas educativas en entornos Web 2.0.*

Carlos Manuel Castaño Garrido, Inmaculada Maiz Olazabalaga, Gorka J. Palazio Arko y José Domingo Villarreal Villamor. 2008. Editorial Síntesis S.A.

[17] *Las nuevas tecnologías en la enseñanza de las ciencias.*

Juan Manuel Muñoz González. 2013. Pixel-Bit: Revista de medios y educación. ISSN 1133-8482. N^o. 42. Páginas 209-210.

[18] *Evaluación del trabajo individual y grupal en un wiki*

Antonio J. Reinoso, Manuel Palomo-Duarte, José Tomás Tocino, Juan Ortega-Valiente y Antonio Balderas. 2015. Congreso Internacional sobre Aprendizaje, Innovación y Competitividad (CINAIC 2015).

[19] *Analysis of the implementation of wiki-based platforms in university education*

Ortega-Valiente, J. Reinoso, A.J. y Muñoz-Mansilla, R. 2013. International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM). MIR Labs. Páginas 41-049.

[20] *Qualitative assessment of wiki-based learning processes*

Antonio Balderas, Manuel Palomo-Duarte, Juan Manuel Dodero e Iván Ruiz Rube. 2012. Proceedings of SPDECE-2012. Ninth multidisciplinary symposium on the design and evaluation of digital content for education. Páginas 161-172.

[21] *Learning-oriented Assessment of Wiki Contributions - How to Assess Wiki Contributions in a Higher Education Learning Setting.*

Emilio J. Rodríguez-Posada, Juan Manuel Doderó, Manuel Palomo-Duarte e Inmaculada Medina-Bulo. 2011. In Proceedings of the 3rd International Conference on Computer Supported Education. Páginas 79-86.

[22] *Indicadores para el seguimiento y evaluación de la competencia de trabajo en equipo a través del método CTMTC*

Ángel Fidalgo Blanco, Dolores Lerís, María Luisa Sein-Echaluce y Francisco J. García-Peñalvo. 2013. II Congreso Internacional sobre Aprendizaje, Innovación y Competitividad (CINAIC).

Anexos

Banco de pruebas unitarias

Amw01_model

get_criteria(\$amw_id, \$user = null, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →array de criterios de evaluación

get_criteria_grades(\$amw_id, \$criteria_id = null, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →array de notas según criterio

get_criteria_average(\$amw_id, \$criteria_id = null, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →nota media según criterio

get_criteria_mode(\$amw_id, \$criteria_id = null, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →moda según criterio

get_user_incoming_grades(\$amw_id, \$user_id, \$criteria_id = null, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →notas recibidas por el usuario
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →notas recibidas por el usuario en el criterio especificado

get_user_incoming_average(\$amw_id, \$user_id, \$criteria_id = null, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →media de notas recibidas por el usuario
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →media de notas recibidas por el usuario en el criterio especificado

get_user_incoming_replies(\$amw_id, \$user_id, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →lista de replies recibidas

get_user_revisors(\$amw_id, \$user_id, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →lista de revisores del usuario

get_user_outgoing_replies(\$amw_id, \$user_id, \$criteria = false, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →replies enviadas por el usuario
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →replies enviadas por el usuario en el criterio especificado

get_general_activity(\$amw_id, \$adate)

- (null, null) →error
- (id no válido, fecha válida) →error
- (id válido, fecha no válida) →error
- (id válido, fecha válida) →actividad general de la wiki

get_average_evolution(\$amw_id, \$user = null, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, null, fecha válida) →evolución de la nota media general
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →evolución de la nota media para el usuario dado

get_group_average_evolution(\$amw_id, \$group_id, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →evolución de la nota media para el grupo dado

get__assessment__lapse(\$amw__id, \$user = null, \$adate)

- (null, null, null) →error
- (id no válido, id válido, fecha válida) →error
- (id válido, id no válido, fecha válida) →error
- (id válido, id válido, fecha no válida) →error
- (id válido, id válido, fecha válida) →lapso de tiempo de evaluación general
- (id válido, null, fecha válida) →lapso de tiempo de evaluación para el usuario dado

get__assessments(\$amw__id, \$wiki__id, \$user = null, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →notas generales
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →notas del usuario dado

get__page__assessments(\$amw__id, \$wiki__id, \$page = null, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →notas generales
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →notas de página dada

get__category__assessments(\$amw__id, \$wiki__id, \$category = null, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →notas generales

- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →notas de la categoría dada

get_group_assessments(\$amw_id, \$wiki_id, \$group = null, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →notas generales
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →notas del grupo dado

get_affected_bytes(\$amw_id, \$wiki_id, \$user = null, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →datos de bytes afectados por notas
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →datos de bytes afectados por notas para el usuario dado

get_group_affected_bytes(\$amw_id, \$wiki_id, \$group = null, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, null, fecha no válida) →datos de bytes afectados por notas
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →datos de bytes afectados por notas para el grupo dado

get_user_radial(\$wiki, \$color, \$user, \$adate)

- (null, null, null, null) →error
- (id no válido, id válido, id válido, fecha válida) →error
- (id válido, id no válido, id válido, fecha válida) →error
- (id válido, id válido, id no válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha no válida) →error
- (id válido, id válido, id válido, fecha válida) →datos de gráfico radial para el usuario dado

get_average(\$color, \$type = null, \$id = null, \$wiki = null, \$adate)

- (null, null, null, null, null) →error
- (id no válido, tipo válido, id válido, id válido fecha válida) →error
- (id válido, tipo no válido, id válido, id válido, fecha válida) →error
- (id válido, tipo válido, id no válido, id válido, fecha válida) →error
- (id válido, tipo válido, id válido, id no válido, fecha válida) →error
- (id válido, tipo válido, id válido, id válido, fecha no válida) →error
- (id válido, tipo válido, id válido, id válido, fecha válida) →media de notas recibidas por el usuario, página, categoría o grupo especificados (dependiendo del tipo) en la wiki especificada

Color_model

cconnection(\$id)

- (null) →error
- (id no válido) →error
- (id válido) →conexión con la base de datos de la fuente cualitativa dada

get_color_type_name(\$id)

- (null) →error
- (id no válido) →error
- (id válido) →tipo de la fuente cualitativa dada

get_color_name(\$id)

- (null) →error
- (id no válido) →error
- (id válido) →nombre de la fuente cualitativa dada

get_color_list(\$user, \$wiki = null)

- (null, null) →error
- (id no válido, id válido) →error
- (id válido, id no válido) →error
- (id válido, null) →lista de fuentes cualitativas del usuario dado
- (id válido) →lista de fuentes cualitativas del usuario dado para la wiki especificada

add_color(\$color_data)

- (null) →error
- (datos no válidos) →error
- (datos válidos) →success

delete_color(\$id)

- (null) →error
- (id no válido) →error
- (id válido) →success

get_color_types()

- () →lista de tipos de fuentes cualitativas del sistema

Connection_model

new_connection(\$server, \$name, \$user, \$password)

- (null, null, null, null) →error
- (servidor inválido, nombre válido, usuario válido, contraseña válida) →error
- (servidor válido, nombre inválido, usuario válido, contraseña válida) →error
- (servidor válido, nombre válido, usuario inválido, contraseña válida) →error
- (servidor válido, nombre válido, usuario válido, contraseña inválida) →error
- (servidor válido, nombre válido, usuario válido, contraseña válida) →id de conexión

connect(\$id)

- (null) →error
- (id válido) →conexión con id dado

delete_connection(\$id)

- (null) →error
- (id válido) →success

Group_model

add_group(\$wiki_id, \$group_name)

- (null, null) →error
- (id válido, nombre inválido) →error
- (id inválido, nombre válido) →error
- (id válido, nombre válido) →success

delete_group(\$group_id)

- (null) →error
- (id inválido) →error
- (id válido) →success

delete_wiki_groups(\$wiki_id)

- (null) →error
- (id inválido) →error
- (id válido) →success

list_groups(\$wiki_id)

- (null) →error
- (id inválido) →error
- (id válido) →lista de grupos de la wiki

group_by_page(\$wiki_id)

- (null) →error
- (id inválido) →error
- (id válido) →success

get_users(\$group_id)

- (null) →error
- (id inválido) →error
- (id válido) →lista de usuarios del grupo

attach_group(\$student, \$group, \$wiki, \$id)

- (null, null, null, null) →error
- (nombre de estudiante inválido, grupo válido, wiki válida, id válido) →error
- (nombre de estudiante válido, grupo inválido, wiki válida, id válido) →error
- (nombre de estudiante válido, grupo válido, wiki inválida, id válido) →error
- (nombre de estudiante válido, grupo válido, wiki válida, id inválido) →error
- (nombre de estudiante válido, grupo válido, wiki válida, id válido) →success

dettach_group(\$student, \$group)

- (null, null) →error
- (nombre de estudiante inválido, grupo válido) →error
- (nombre de estudiante válido, grupo inválido) →error
- (nombre de estudiante válido, grupo válido) →success

get_group_name(\$id)

- (null) →error
- (id inválido) →error
- (id válido) →nombre del grupo

Http_model

answer(\$data)

- (null) →error
- (datos inválidos) →error
- (datos válidos) →undefined

Report_model

get_analysis_date(\$id)

- (null) →error
- (id inválido) →error
- (id válido) →fecha del análisis dado

get_public_analysis_date(\$id)

- (null) →error
- (id inválido) →error
- (id válido) →fecha de publicación del análisis dado

get_analysis_list(\$user)

- (null) →error
- (id inválido) →error
- (id válido) →lista de análisis del usuario dado

delete_analysis(\$analysis)

- (null) →error
- (id inválido) →error
- (id válido) →success

Security_model

php_full_sanitize(\$input_data)

- (null) →error
- (array) →datos de entrada sanitizados

filter_post(\$input_data)

- (null) →error
- (array) →datos de entrada filtrados

encrypt_decrypt(\$action, \$string)

- (null, null) →error
- (acción inválida, string) →error
- (acción válida, string) →string de entrada encriptado o desencriptado según opción

check_method_permissions(\$class, \$method)

- (null, null) →error
- (clase inválida, método válido) →error
- (clase válida, método inválido) →error
- (clase válida con permisos, método válido sin permisos) →error

- (clase válida sin permisos, método válido con permisos) →error
- (clase válida con permisos, método válido con permisos) →undefined

Session_model

update_last_session(\$uname)

- (usuario correcto) →sesión actualizada
- (usuario incorrecto) →error
- (null) →error

login(\$uname, \$pass)

- (usuario correcto, contraseña correcta) →sesión actualizada
- (usuario correcto, contraseña incorrecta) →error
- (usuario incorrecto, contraseña correcta) →error
- (null, null) →error

is_logged()

- (usuario autenticado) →true
- (usuario no autenticado) →false

is_admin()

- (usuario es administrador) →true
- (usuario no es administrador) →false

logout()

- (usuario autenticado) →usuario no autenticado
- (usuario no autenticado) →usuario no autenticado

set(\$data)

- (vacío/null) →nada guardado en caché
- (datos válidos) →datos guardados en caché

User_model

get_user_list(\$discardme = false)

- (null) →array de usuarios del sistema
- (true) →array de usuarios del sistema sin el usuario autenticado
- (false) →array de usuarios del sistema

get_admin()

- () → usuario admin

get_user_data(\$user)

- (usuario válido) → array de información sobre el usuario dado
- (usuario no válido) → error
- (null) → error

change_password(\$user, \$new_password)

- (usuario válido, contraseña válida) → contraseña cambiada
- (usuario inválido, contraseña válida) → error
- (usuario válido, contraseña inválida) → error
- (null, null) → error

get_old_password(\$user)

- (usuario válido) → contraseña antigua del usuario, cifrada
- (usuario no válido) → error
- (null) → error

add_user(\$user_data)

- (datos válidos) → usuario añadido con los datos dados
- (datos no válidos) → error
- (null) → error

delete_user(\$user_id)

- (usuario válido) → usuario borrado
- (usuario no válido) → error
- (null) → error

get_parameters()

- () → array de parámetros registrados en el sistema

save_parameters(\$thresholds, \$quality_function = null)

- (límites válidos, función válida) →parámetros guardados correctamente
- (límites válidos, función no válida) →error
- (límites válidos, null) →parámetros guardados correctamente
- (límites no válidos, función válida) →error
- (límites no válidos, función no válida) →error
- (límites no válidos, null) →error
- (null, función válida) →error
- (null, null) →error

get_user_thresholds(\$user)

- (usuario válido) →parámetros guardados para el usuario dado
- (usuario no válido) →error
- (null) →error

get_quality_function()

- () →función de calidad asignada al usuario autenticado

Wiki_model

wconnection(\$id)

- (id válido) →conexión con la wiki dada
- (id no válido) →error
- (null) →error

get_wiki_list(\$user)

- (usuario válido) →lista de wikis asociadas al usuario dado
- (usuario no válido) →error
- (null) →error

delete_wiki(\$id)

- (id válido) →wiki dada borrada
- (id no válido) →error
- (null) →error

get_wiki_name(\$id)

- (id válido) → nombre de la wiki dada
- (id no válido) → error
- (null) → error

get_wiki_students(\$id)

- (id válido) → lista de estudiantes de la wiki dada
- (id no válido) → error
- (null) → error

get_groups_model()

- () → clase del modelo de grupos a utilizar

get_group_members(\$wiki_id, \$group)

- (id válido, grupo válido) → lista de miembros del grupo especificado
- (id no válido, grupo válido) → error
- (id válido, grupo no válido) → error
- (null, null) → error

get_group_evolution(\$wiki_id, \$analysis, \$group)

- (id válido, análisis válido, grupo válido) → array con datos de evolución del grupo
- (id válido, análisis válido, grupo no válido) → error
- (id válido, análisis no válido, grupo válido) → error
- (id válido, análisis no válido, grupo no válido) → error
- (id no válido, análisis válido, grupo válido) → error
- (id no válido, análisis válido, grupo no válido) → error
- (id no válido, análisis no válido, grupo válido) → error
- (id no válido, análisis no válido, grupo no válido) → error
- (null, null, null) → error

get_group_percentages(\$wiki_id, \$analysis, \$group)

- (id válido, análisis válido, grupo válido) →array con datos de porcentajes del grupo
- (id válido, análisis válido, grupo no válido) →error
- (id válido, análisis no válido, grupo válido) →error
- (id válido, análisis no válido, grupo no válido) →error
- (id no válido, análisis válido, grupo válido) →error
- (id no válido, análisis válido, grupo no válido) →error
- (id no válido, análisis no válido, grupo válido) →error
- (id no válido, análisis no válido, grupo no válido) →error
- (null, null, null) →error

get_group_activity(\$wiki_id, \$analysis, \$group)

- (id válido, análisis válido, grupo válido) →array con datos de actividad del grupo
- (id válido, análisis válido, grupo no válido) →error
- (id válido, análisis no válido, grupo válido) →error
- (id válido, análisis no válido, grupo no válido) →error
- (id no válido, análisis válido, grupo válido) →error
- (id no válido, análisis válido, grupo no válido) →error
- (id no válido, análisis no válido, grupo válido) →error
- (id no válido, análisis no válido, grupo no válido) →error
- (null, null, null) →error

get_group_uploads(\$wiki_id, \$analysis, \$group)

- (id válido, análisis válido, grupo válido) →array con datos de subidas de archivos del grupo
- (id válido, análisis válido, grupo no válido) →error
- (id válido, análisis no válido, grupo válido) →error
- (id válido, análisis no válido, grupo no válido) →error
- (id no válido, análisis válido, grupo válido) →error
- (id no válido, análisis válido, grupo no válido) →error
- (id no válido, análisis no válido, grupo válido) →error
- (id no válido, análisis no válido, grupo no válido) →error
- (null, null, null) →error

get_group_pages(\$wiki_id, \$analysis, \$group)

- (id válido, análisis válido, grupo válido) →array con datos de páginas del grupo
- (id válido, análisis válido, grupo no válido) →error
- (id válido, análisis no válido, grupo válido) →error
- (id válido, análisis no válido, grupo no válido) →error
- (id no válido, análisis válido, grupo válido) →error
- (id no válido, análisis válido, grupo no válido) →error
- (id no válido, análisis no válido, grupo válido) →error
- (id no válido, análisis no válido, grupo no válido) →error
- (null, null, null) →error

get_group_categories(\$wiki_id, \$analysis, \$group)

- (id válido, análisis válido, grupo válido) →array con datos de categorías del grupo
- (id válido, análisis válido, grupo no válido) →error
- (id válido, análisis no válido, grupo válido) →error
- (id válido, análisis no válido, grupo no válido) →error
- (id no válido, análisis válido, grupo válido) →error
- (id no válido, análisis válido, grupo no válido) →error
- (id no válido, análisis no válido, grupo válido) →error
- (id no válido, análisis no válido, grupo no válido) →error
- (null, null, null) →error

get_user_evolution(\$wiki_id, \$analysis, \$user)

- (id válido, análisis válido, usuario válido) →array con datos de evolución del usuario
- (id válido, análisis válido, usuario no válido) →error
- (id válido, análisis no válido, usuario válido) →error
- (id válido, análisis no válido, usuario no válido) →error
- (id no válido, análisis válido, usuario válido) →error
- (id no válido, análisis válido, usuario no válido) →error
- (id no válido, análisis no válido, usuario válido) →error
- (id no válido, análisis no válido, usuario no válido) →error
- (null, null, null) →error

get_user_percentages(\$wiki_id, \$analysis, \$user)

- (id válido, análisis válido, usuario válido) →array con datos de porcentajes del usuario
- (id válido, análisis válido, usuario no válido) →error
- (id válido, análisis no válido, usuario válido) →error
- (id válido, análisis no válido, usuario no válido) →error
- (id no válido, análisis válido, usuario válido) →error
- (id no válido, análisis válido, usuario no válido) →error
- (id no válido, análisis no válido, usuario válido) →error
- (id no válido, análisis no válido, usuario no válido) →error
- (null, null, null) →error

get_user_activity(\$wiki_id, \$analysis, \$user)

- (id válido, análisis válido, usuario válido) →array con datos de actividad del usuario
- (id válido, análisis válido, usuario no válido) →error
- (id válido, análisis no válido, usuario válido) →error
- (id válido, análisis no válido, usuario no válido) →error
- (id no válido, análisis válido, usuario válido) →error
- (id no válido, análisis válido, usuario no válido) →error
- (id no válido, análisis no válido, usuario válido) →error
- (id no válido, análisis no válido, usuario no válido) →error
- (null, null, null) →error

get_user_uploads(\$wiki_id, \$analysis, \$user)

- (id válido, análisis válido, usuario válido) →array con datos de subidas de archivo del usuario
- (id válido, análisis válido, usuario no válido) →error
- (id válido, análisis no válido, usuario válido) →error
- (id válido, análisis no válido, usuario no válido) →error
- (id no válido, análisis válido, usuario válido) →error
- (id no válido, análisis válido, usuario no válido) →error
- (id no válido, análisis no válido, usuario válido) →error
- (id no válido, análisis no válido, usuario no válido) →error
- (null, null, null) →error

get_user_pages(\$wiki_id, \$analysis, \$user)

- (id válido, análisis válido, usuario válido) →array con datos de páginas del usuario
- (id válido, análisis válido, usuario no válido) →error
- (id válido, análisis no válido, usuario válido) →error
- (id válido, análisis no válido, usuario no válido) →error
- (id no válido, análisis válido, usuario válido) →error
- (id no válido, análisis válido, usuario no válido) →error
- (id no válido, análisis no válido, usuario válido) →error
- (id no válido, análisis no válido, usuario no válido) →error
- (null, null, null) →error

get_user_categories(\$wiki_id, \$analysis, \$user)

- (id válido, análisis válido, usuario válido) →array con datos de categorías del usuario
- (id válido, análisis válido, usuario no válido) →error
- (id válido, análisis no válido, usuario válido) →error
- (id válido, análisis no válido, usuario no válido) →error
- (id no válido, análisis válido, usuario válido) →error
- (id no válido, análisis válido, usuario no válido) →error
- (id no válido, análisis no válido, usuario válido) →error
- (id no válido, análisis no válido, usuario no válido) →error
- (null, null, null) →error

get_user_name(\$wiki_id, \$id)

- (id válido, id válido) →nombre del usuario
- (id válido, id no válido) →error
- (id no válido, id válido) →error
- (id no válido, id no válido) →error
- (null, null) →error

get_user_groups(\$wiki_id, \$user)

- (id válido, usuario válido) → lista de grupos del usuario
- (id válido, usuario no válido) → error
- (id no válido, usuario válido) → error
- (id no válido, usuario no válido) → error
- (null, null) → error

get_category_evolution(\$wiki_id, \$analysis, \$category)

- (id válido, análisis válido, categoría válida) → array con datos de evolución de la categoría
- (id válido, análisis válido, categoría no válida) → error
- (id válido, análisis no válido, categoría válida) → error
- (id válido, análisis no válido, categoría no válida) → error
- (id no válido, análisis válido, categoría válida) → error
- (id no válido, análisis válido, categoría no válida) → error
- (id no válido, análisis no válido, categoría válida) → error
- (id no válido, análisis no válido, categoría no válida) → error
- (null, null, null) → error

get_category_activity(\$wiki_id, \$analysis, \$category)

- (id válido, análisis válido, categoría válida) → array con datos de actividad de la categoría
- (id válido, análisis válido, categoría no válida) → error
- (id válido, análisis no válido, categoría válida) → error
- (id válido, análisis no válido, categoría no válida) → error
- (id no válido, análisis válido, categoría válida) → error
- (id no válido, análisis válido, categoría no válida) → error
- (id no válido, análisis no válido, categoría válida) → error
- (id no válido, análisis no válido, categoría no válida) → error
- (null, null, null) → error

get_category_users(\$wiki_id, \$analysis, \$category)

- (id válido, análisis válido, categoría válida) →array con datos de usuarios de la categoría
- (id válido, análisis válido, categoría no válida) →error
- (id válido, análisis no válido, categoría válida) →error
- (id válido, análisis no válido, categoría no válida) →error
- (id no válido, análisis válido, categoría válida) →error
- (id no válido, análisis válido, categoría no válida) →error
- (id no válido, análisis no válido, categoría válida) →error
- (id no válido, análisis no válido, categoría no válida) →error
- (null, null, null) →error

get_category_pages(\$wiki_id, \$analysis, \$category)

- (id válido, análisis válido, categoría válida) →array con datos de páginas de la categoría
- (id válido, análisis válido, categoría no válida) →error
- (id válido, análisis no válido, categoría válida) →error
- (id válido, análisis no válido, categoría no válida) →error
- (id no válido, análisis válido, categoría válida) →error
- (id no válido, análisis válido, categoría no válida) →error
- (id no válido, análisis no válido, categoría válida) →error
- (id no válido, análisis no válido, categoría no válida) →error
- (null, null, null) →error

get_page_evolution(\$wiki_id, \$analysis, \$page)

- (id válido, análisis válido, página válida) →array con datos de evolución de la página
- (id válido, análisis válido, página no válida) →error
- (id válido, análisis no válido, página válida) →error
- (id válido, análisis no válido, página no válida) →error
- (id no válido, análisis válido, página válida) →error
- (id no válido, análisis válido, página no válida) →error
- (id no válido, análisis no válido, página válida) →error
- (id no válido, análisis no válido, página no válida) →error
- (null, null, null) →error

get_page_percentages(\$wiki_id, \$analysis, \$page)

- (id válido, análisis válido, página válida) →array con datos de porcentajes de la página
- (id válido, análisis válido, página no válida) →error
- (id válido, análisis no válido, página válida) →error
- (id válido, análisis no válido, página no válida) →error
- (id no válido, análisis válido, página válida) →error
- (id no válido, análisis válido, página no válida) →error
- (id no válido, análisis no válido, página válida) →error
- (id no válido, análisis no válido, página no válida) →error
- (null, null, null) →error

get_page_activity(\$wiki_id, \$analysis, \$page)

- (id válido, análisis válido, página válida) →array con datos de actividad de la página
- (id válido, análisis válido, página no válida) →error
- (id válido, análisis no válido, página válida) →error
- (id válido, análisis no válido, página no válida) →error
- (id no válido, análisis válido, página válida) →error
- (id no válido, análisis válido, página no válida) →error
- (id no válido, análisis no válido, página válida) →error
- (id no válido, análisis no válido, página no válida) →error
- (null, null, null) →error

get_page_uploads(\$wiki_id, \$analysis, \$page)

- (id válido, análisis válido, página válida) →array con datos de subidas de archivos de la página
- (id válido, análisis válido, página no válida) →error
- (id válido, análisis no válido, página válida) →error
- (id válido, análisis no válido, página no válida) →error
- (id no válido, análisis válido, página válida) →error
- (id no válido, análisis válido, página no válida) →error
- (id no válido, análisis no válido, página válida) →error
- (id no válido, análisis no válido, página no válida) →error
- (null, null, null) →error

get__page__users(\$wiki_id, \$analysis, \$page)

- (id válido, análisis válido, página válida) →array con datos de usuarios de la página
- (id válido, análisis válido, página no válida) →error
- (id válido, análisis no válido, página válida) →error
- (id válido, análisis no válido, página no válida) →error
- (id no válido, análisis válido, página válida) →error
- (id no válido, análisis válido, página no válida) →error
- (id no válido, análisis no válido, página válida) →error
- (id no válido, análisis no válido, página no válida) →error
- (null, null, null) →error

get__page__categories(\$wiki_id, \$analysis, \$page)

- (id válido, análisis válido, página válida) →array con datos de categorías de la página
- (id válido, análisis válido, página no válida) →error
- (id válido, análisis no válido, página válida) →error
- (id válido, análisis no válido, página no válida) →error
- (id no válido, análisis válido, página válida) →error
- (id no válido, análisis válido, página no válida) →error
- (id no válido, análisis no válido, página válida) →error
- (id no válido, análisis no válido, página no válida) →error
- (null, null, null) →error

get__general__evolution(\$wiki_id, \$analysis)

- (id válido, análisis válido) →array con datos de evolución de la wiki
- (id válido, análisis no válido) →error
- (id no válido, análisis válido) →error
- (id no válido, análisis no válido) →error
- (null, null, null) →error

get_general_percentages(\$wiki_id, \$analysis)

- (id válido, análisis válido) →array con datos de porcentajes de la wiki
- (id válido, análisis no válido) →error
- (id no válido, análisis válido) →error
- (id no válido, análisis no válido) →error
- (null, null, null) →error

get_general_activity(\$wiki_id, \$analysis)

- (id válido, análisis válido) →array con datos de actividad de la wiki
- (id válido, análisis no válido) →error
- (id no válido, análisis válido) →error
- (id no válido, análisis no válido) →error
- (null, null, null) →error

get_general_uploads(\$wiki_id, \$analysis)

- (id válido, análisis válido) →array con datos de subida de archivos de la wiki
- (id válido, análisis no válido) →error
- (id no válido, análisis válido) →error
- (id no válido, análisis no válido) →error
- (null, null, null) →error

get_general_users(\$wiki_id, \$analysis)

- (id válido, análisis válido) →array con datos de usuarios de la wiki
- (id válido, análisis no válido) →error
- (id no válido, análisis válido) →error
- (id no válido, análisis no válido) →error
- (null, null, null) →error

namespace_string(\$namespace_code)

- (código de namespace válido) →cadena de namespace
- (código de namespace no válido) →error
- (null) →error

get_general_pages(\$wiki_id, \$analysis)

- (id válido, análisis válido) →array con datos de páginas de la wiki
- (id válido, análisis no válido) →error
- (id no válido, análisis válido) →error
- (id no válido, análisis no válido) →error
- (null, null, null) →error

get_general_categories(\$wiki_id, \$analysis)

- (id válido, análisis válido) →array con datos de categorías de la wiki
- (id válido, análisis no válido) →error
- (id no válido, análisis válido) →error
- (id no válido, análisis no válido) →error
- (null, null, null) →error

get_general_groups(\$wiki_id)

- (id válido) →lista de grupos de la wiki dada
- (id no válido) →error
- (null) →error

check_threshold(\$threshold, \$value, \$wiki, \$analysis, \$filter = false)

- (límite válido, valor, wiki válida, análisis válido, false) →check del valor con respecto al límite
- (límite válido, valor, wiki válida, análisis válido, true) →check del valor con respecto al límite, filtrado por el objeto dado
- (límite válido, valor, wiki válida, análisis no válido, false) →error
- (límite válido, valor, wiki válida, análisis no válido, filtro) →error
- (límite válido, valor, wiki no válida, análisis válido, false) →error
- (límite válido, valor, wiki no válida, análisis válido, filtro) →error
- (límite válido, valor, wiki no válida, análisis no válido, false) →error
- (límite válido, valor, wiki no válida, análisis no válido, filtro) →error
- (límite no válido, valor, wiki válida, análisis válido, false) →error
- (límite no válido, valor, wiki válida, análisis válido, filtro) →error

- (límite no válido, valor, wiki válida, análisis no válido, false) →error
- (límite no válido, valor, wiki válida, análisis no válido, filtro) →error
- (límite no válido, valor, wiki no válida, análisis válido, false) →error
- (límite no válido, valor, wiki no válida, análisis válido, filtro) →error
- (límite no válido, valor, wiki no válida, análisis no válido, false) →error
- (límite no válido, valor, wiki no válida, análisis no válido, filtro) →error

get_warnings(\$wiki, \$datetime, \$type = 'wiki', \$filter)

- (wiki válida, fecha, 'wiki', false) →listado de umbrales configurados para la wiki dada en la fecha especificada
- (wiki válida, fecha, 'wiki', filtro) →error
- (wiki no válida, fecha, 'wiki', false) →error
- (wiki no válida, fecha, 'wiki', filtro) →error

add_wiki(\$wiki_data)

- (datos válidos) →wiki guardada correctamente
- (datos no válidos) →error
- (null) →error

analyze_wiki(\$wiki, \$color)

- (wiki válida, color válido) →wiki analizada usando la fuente cualitativa especificada
- (wiki válida, null) →wiki analizada
- (wiki válida, color no válido) →error
- (wiki no válida, color válido) →error
- (wiki no válida, color no válido) →error
- (null, null) →error

get_student_name(\$wiki, \$id)

- (wiki válida, id de estudiante válido) →nombre del estudiante
- (wiki válida, id de estudiante no válido) →error
- (wiki no válida, id de estudiante válido) →error
- (wiki no válida, id de estudiante no válido) →error
- (null, null) →error

get_page_name(\$wiki, \$id)

- (wiki válida, id de página válido) →nombre de la página
- (wiki válida, id de página no válido) →error
- (wiki no válida, id de página válido) →error
- (wiki no válida, id de página no válido) →error
- (null, null) →error

get_category_name(\$wiki, \$id)

- (wiki válida, id de categoría válido) →nombre de la categoría
- (wiki válida, id de categoría no válido) →error
- (wiki no válida, id de categoría válido) →error
- (wiki no válida, id de categoría no válido) →error
- (null, null) →error

Banco de pruebas funcionales

Sesión - Login - Caso positivo

Precondiciones:

Usuario con acceso al sistema. Pantalla de login.

Pasos a seguir:

1. Escribir usuario correcto.
2. Escribir contraseña correcta.
3. Click Submit

Resultado:

El usuario accede al sistema y se le presenta su lista de reportes.

Sesión - Login - Caso negativo

Precondiciones:

Usuario con acceso al sistema. Pantalla de login.

Pasos a seguir:

1. Escribir usuario correcto.
2. Escribir contraseña incorrecta.
3. Click Submit.

Resultado:

Un mensaje de error aparece indicando que las credenciales no son correctas.

Sesión - Salir - Caso positivo

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en el menú desplegable superior derecho.
2. Hacer click en cerrar sesión.

Resultado:

Usuario fuera de la aplicación, en la pantalla de login.

Cuenta - Cambiar contraseña - Caso positivo

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en el menú desplegable superior derecho.
2. Hacer click en cambiar contraseña.
3. Escribir la contraseña antigua, la nueva y confirmar.
4. Click en guardar.

Resultado:

Contraseña de usuario cambiada correctamente.

Cuenta - Cambiar contraseña - Caso negativo

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en el menú desplegable superior derecho.
2. Hacer click en cambiar contraseña.
3. Escribir la contraseña antigua de forma incorrecta, la nueva y confirmar.
4. Click en guardar.

Resultado:

Un error aparece informando de que la contraseña antigua no es correcta.

Reportes - Ver lista de reportes - Caso positivo

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.

Resultado:

El listado de reportes del usuario se muestra en pantalla.

Reportes - Añadir reporte - Caso positivo

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.
2. Hacer click en Añadir Reporte.
3. Seleccionar wiki.
4. Click en analizar.

Resultado:

Un nuevo reporte aparece en la lista, con la wiki seleccionada y un aspa en el campo “fuente cualitativa”.

Reportes - Añadir reporte - Caso positivo con fuente cualitativa

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.
2. Hacer click en Añadir Reporte.
3. Seleccionar wiki y fuente cualitativa.
4. Click en analizar.

Resultado:

Un nuevo reporte aparece en la lista, con la wiki y fuente cualitativa seleccionadas.

Reportes - Añadir reporte - Caso negativo

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.
2. Hacer click en Añadir Reporte.
3. Seleccionar wiki y fuente cualitativa.
4. Click en cancelar.

Resultado:

No se añade ningún reporte a la lista.

Reportes - Ver reporte - General - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.
2. Elegir un reporte de la lista y hacer click en “Ver”.

Resultado:

Se muestra el reporte general de la wiki.

Reportes - Ver reporte - Reporte de alumno - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.
2. Elegir un reporte de la lista y hacer click en “Ver”.
3. Ir a la tabla de usuarios.
4. Elegir un usuario de la tabla y hacer click en su nombre.

Resultado:

Se muestra el reporte de usuario.

Reportes - Ver reporte - Reporte de página - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.
2. Elegir un reporte de la lista y hacer click en “Ver”.
3. Ir a la tabla de páginas.
4. Elegir una página de la tabla y hacer click en su nombre.

Resultado:

Se muestra el reporte de página.

Reportes - Ver reporte - Reporte de categoría - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.
2. Elegir un reporte de la lista y hacer click en “Ver”.
3. Ir a la tabla de categorías.
4. Elegir una categoría de la tabla y hacer click en su nombre.

Resultado:

Se muestra el reporte de página.

Reportes - Ver reporte - Reporte de grupo - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Reportes.
2. Elegir un reporte de la lista y hacer click en “Ver”.
3. Ir a la tabla de grupos.
4. Elegir un grupo de la tabla y hacer click en su nombre.

Resultado:

Se muestra el reporte de grupo.

Configurar - Opciones - Umbrales de aviso - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Opciones.
3. Establecer el primer y último slide a la mitad de su valor máximo.
4. Click en Guardar.
5. Hacer click en Reportes.
6. Hacer click en Configuración.
7. Ir a la pestaña Opciones.

Resultado:

Los slides se mantienen en la posición guardada previamente.

Configurar - Opciones - Función de calidad - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Editar como administrador el fichero de configuración de funciones de calidad. Añadir una función igual a la función por defecto con el nombre FUNCION1.
2. Iniciar sesión en la aplicación como profesor.
3. Hacer click en Configuración.
4. Ir a la pestaña Opciones.
5. Establecer la función de calidad a FUNCION1.
6. Click en Guardar.
7. Hacer click en Reportes.
8. Hacer click en Configuración.
9. Ir a la pestaña Opciones.

Resultado:

La función de calidad guardada previamente se mantiene.

Configurar - Opciones - Función de calidad - Caso negativo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Opciones.
3. Establecer la función de calidad a FUNCION2.
4. Click en Guardar.

Resultado:

Aparece un mensaje de error indicando que la función de calidad no existe en el sistema.

Configurar - Wikis - Añadir wiki - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Wikis.
3. Hacer click en Añadir Wiki.
4. Rellenar los campos del formulario con datos válidos de conexión con una wiki.
5. Hacer click en Guardar.
6. Ir a Reportes.
7. Hacer click en Añadir Reporte.

Resultado:

La nueva wiki está disponible en el desplegable de wikis.

Configurar - Wikis - Añadir wiki - Caso negativo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Wikis.
3. Hacer click en Añadir Wiki.
4. Rellenar los campos del formulario con datos no válidos de conexión con una wiki.
5. Hacer click en Guardar.

Resultado:

Aparece un mensaje de error indicando que no fue posible conectar con la wiki.

Configurar - Wikis - Borrar wiki - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Wikis.
3. Elegir una wiki y hacer click en Borrar.
4. Confirmar el borrado.

Resultado:

La wiki desaparece de la lista y deja de estar disponible para generar reportes.

Configurar - Wikis - Gestionar grupos - Añadir grupos - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Wikis.
3. Elegir una wiki y hacer click en Gestionar Grupos.
4. Hacer click en Añadir Grupo.
5. Escribir GRUPO1 como nombre.
6. Hacer click en Guardar.

Resultado:

El grupo se guardar y está disponible para asignar.

Configurar - Wikis - Gestionar grupos - Asignar grupo - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Wikis.
3. Elegir una wiki y hacer click en Gestionar Grupos.
4. Elegir un alumno y hacer click en Asignar grupo.
5. Seleccionar el grupo GRUPO1 del desplegable.
6. Hacer click en Asignar.

Resultado:

El grupo queda asignado al alumno y aparece junto a su nombre.

Configurar - Wikis - Gestionar grupos - Desasignar grupo - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Wikis.
3. Elegir una wiki y hacer click en Gestionar Grupos.
4. Elegir un alumno con grupos asignados y hacer click en el aspa roja que aparece junto al primer grupo.

Resultado:

El grupo se desasigna y desaparece de la derecha del nombre del alumno.

Configurar - Wikis - Gestionar grupos - Asignación automática de grupos - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Wikis.
3. Elegir una wiki y hacer click en Gestionar Grupos.
4. Hacer click en Asignación Automática de Grupos.

Resultado:

Los grupos se asignan a los alumnos de forma automática, tomando como criterio la página en la que han trabajado.

Configurar - Fuentes cualitativas - Añadir fuente cualitativa - Caso positivo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Fuentes cualitativas.
3. Hacer click en Añadir fuente cualitativa.
4. Rellenar los campos del formulario con datos válidos de conexión con una fuente cualitativa.
5. Hacer click en Guardar.
6. Ir a Reportes.
7. Hacer click en Añadir Reporte.

Resultado:

La fuente queda registrada y está disponible para seleccionar en el desplegable de fuentes cualitativas, seleccionando antes la wiki a la que se asignó.

Configurar - Fuentes cualitativas - Añadir fuente cualitativa - Caso negativo**Precondiciones:**

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Fuentes cualitativas.
3. Hacer click en Añadir fuente cualitativa.
4. Rellenar los campos del formulario con datos no válidos de conexión con una fuente cualitativa.

Hacer click en Guardar.

Resultado:

El sistema muestra un error indicando que no se pudo conectar con la fuente cualitativa.

Configurar - Fuentes cualitativas - Borrar fuente cualitativa - Caso positivo

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Fuentes cualitativas.
3. Elegir una fuente cualitativa y hacer click en Borrar.
4. Confirmar el borrado.

Resultado:

La fuente queda borrada de la lista y no está disponible para usarla en los reportes.

(Admin) Configurar - Usuarios - Ver lista de usuarios - Caso positivo

Precondiciones:

Administrador dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Usuarios.

Resultado:

Se muestra una lista de todos los usuarios del sistema.

(Admin) Configurar - Usuarios - Añadir usuario - Caso positivo

Precondiciones:

Administrador dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Usuarios.
3. Hacer click en Añadir usuario.
4. Rellenar el formulario con datos válidos.
5. Hacer click en Añadir.

Resultado:

Un nuevo usuario se registra en el sistema y sus credenciales están listas para iniciar sesión.

(Admin) Configurar - Usuarios - Borrar usuario - Caso positivo**Precondiciones:**

Administrador dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Configuración.
2. Ir a la pestaña Usuarios.
3. Elegir un usuario de la lista distinto al actual.
4. Hacer click en Borrar.
5. Confirmar borrado.

Resultado:

El usuario queda borrado del sistema y no puede iniciar sesión más.

(Admin) Configurar - Añadir función de calidad - Caso positivo**Precondiciones:**

Administrado dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Editar el fichero de configuración de funciones de calidad.
2. Añadir una nueva función de calidad llamada FUNCION1.
3. Guardar el fichero y cerrarlo.
4. Ir a Configuración.
5. Ir a la pestaña de Opciones.
6. Escribir FUNCION1 en el campo de Función de calidad.
7. Hacer click en guardar.

Resultado:

La función se guarda correctamente.

(Admin) Configurar - Añadir datos de contacto - Caso positivo

Precondiciones:

Administrador dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Editar el fichero de datos de contacto.
2. Añadir datos reales a las variables presentes.
3. Guardar el fichero y cerrarlo.
4. Hacer click en Contacto.

Resultado:

Los datos de contacto escritos en el fichero se reflejan en la aplicación.

Contacto - Ver información de contacto - Caso positivo

Precondiciones:

Usuario dentro de la aplicación. Cualquier pantalla.

Pasos a seguir:

1. Hacer click en Contacto.

Resultado:

Los datos de contacto con el administrador y el soporte técnico se muestran en pantalla.

Diagrama de Gantt. Desarrollo estimado.

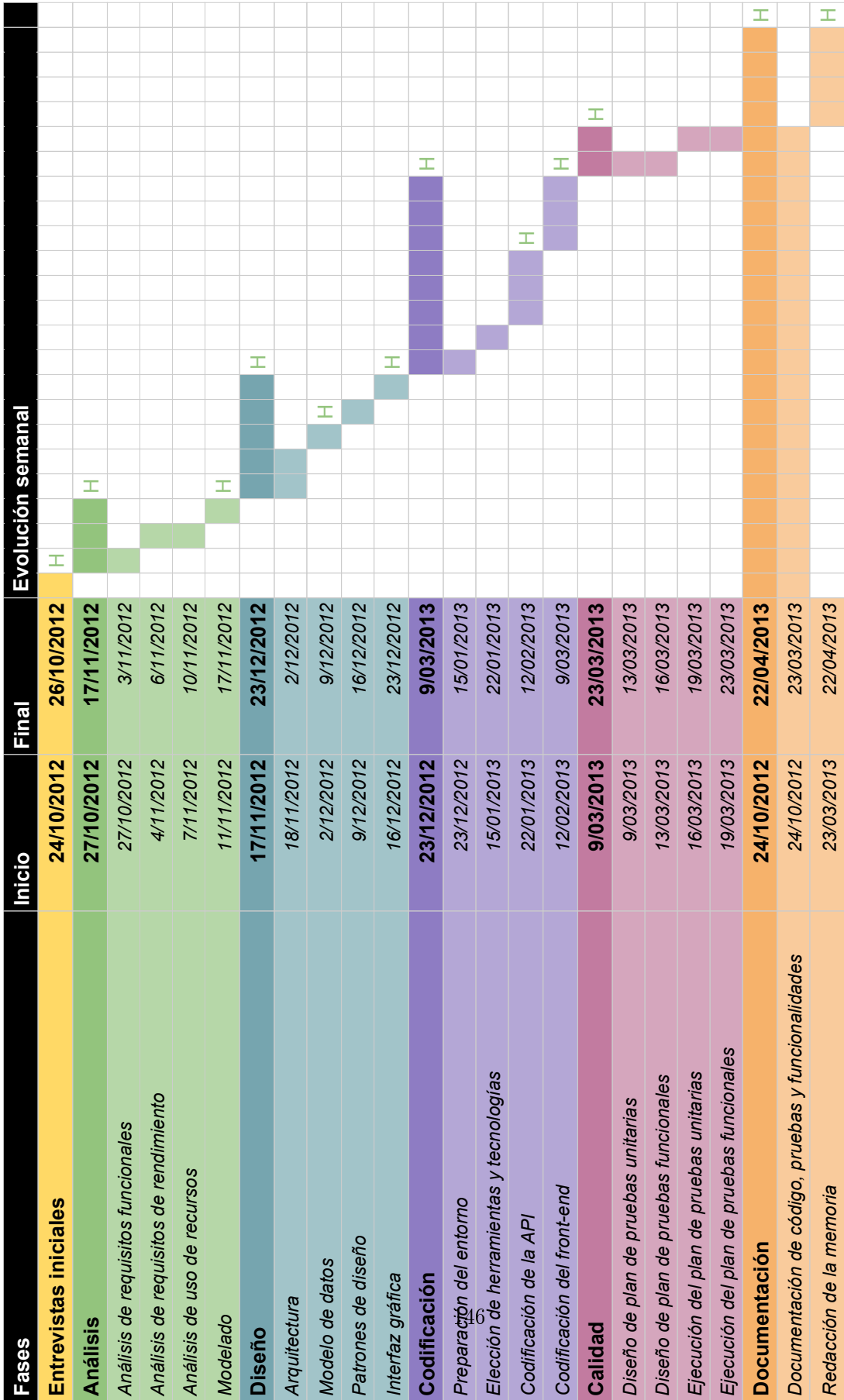
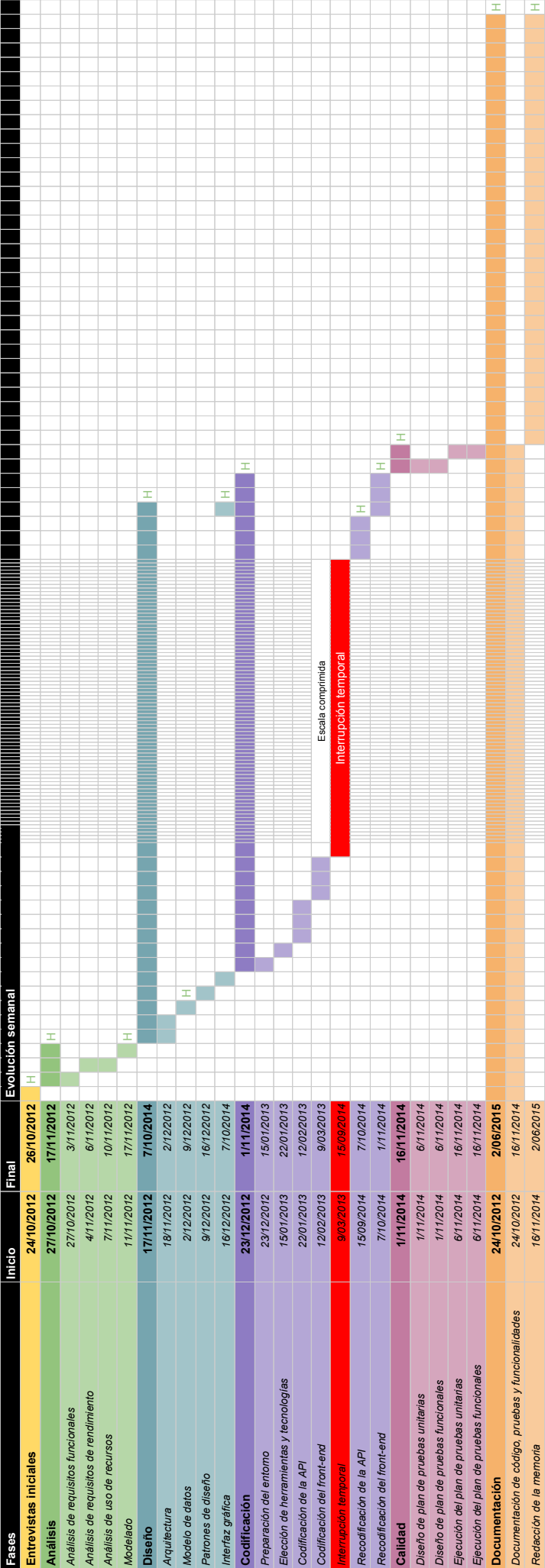


Diagrama de Gantt. Desarrollo real.



Información sobre Licencia

NOTE: This license covers CleverFigures code files, developed by Universidad de Cádiz, which also HAVE the Apache license header, but NOT any of the third-party libraries or frameworks used to develop it. To get more information about these other licenses, please, check their corresponding website or source code.

Apache License Version 2.0, January 2004 <http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

'License' shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

'Licensor' shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

'Legal Entity' shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, 'control' means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50 %) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

'You' (or 'Your') shall mean an individual or Legal Entity exercising permissions granted by this License.

'Source' form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

'Object' form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

'Work' shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

'Derivative Works' shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

'Contribution' shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, 'submitted' means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as 'Not a Contribution.'

'Contributor' shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

- d) If the Work includes a 'NOTICE' text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an 'AS IS' BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer

failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS